

SCIENTIFIC PROGRAMME

KEYNOTE SPEECH

KNOWLEDGE DISCOVERY IN DATABASES AND MODELLING, HOW DO THEY FIT TOGETHER?

Danny F. Van Welden
Applied Mathematics, Biometrics and Process Control
Ghent University
B-9000 Gent
Belgium
Email: Danny.VanWelden@rug.ac.be

INTRODUCTION

The past decades have seen a dramatic increase in the amount of data being stored. This has become possible by the availability of large amounts of computing power and storage at low cost. Nevertheless, the resulting mountains of data contain a lot of unexploited information nuggets waiting to be mined. These information nuggets play a key role in the competition among companies and in new discoveries in science. Although statistics, query tools and OLAP (On Line Analytical Processing) tools are no longer strangers to management they do not contain in essence the potential of KDD (Knowledge Discovery in Databases).

This paper gives an overview of KDD and touches on its link with General Systems Theory (GST). It is very difficult to describe the commonality and the emphasis shifts in a few pages. It is assumed that the reader is already familiar with system theory and simulation.

It can be shown that the life cycles of KDD and GST are almost identical, but with some emphasis shifts. Especially with regard to the sub-field of data mining, and put in the right perspective to KDD, it can be shown that data mining is nothing else but 'classical' modelling. The top-down and bottom-up approach to modelling will be compared to the deductive and inductive approach in KDD. Because of the abstraction level of GST, one expects that KDD can learn from GST, but an opposite knowledge transfer is equally possible. KDD puts more emphasis on pre-processing of data and the goal settings are more elaborated. These emphases can be transferred to the domain of general systems theory, therefore putting the latter in an even broader perspective.

KNOWLEDGE DISCOVERY IN DATABASES AND DATA MINING

Nowadays, more and more data is digitised and stored in databases. This happens thanks to automation in generating and collecting of data; via bar code readers, chip cards, etc. Tax returns, health care transactions, consumer behaviour, all kinds of commercial transactions, industrial

transactions, scientific information, exploitation data, etc., are stored in databases, which consequently have grown explosively during the last decades. Unfortunately, the resulting massive amount of facts cannot be processed by humans alone anymore to unravel informative hidden patterns. As a consequence, tools and techniques have emerged that can assist humans intelligently and automatically in analysing the abundant amount of raw data to obtain the desired information. This kind of exploitation of data constitutes the domain of Knowledge Discovery in Databases (KDD).

Generally, KDD is a data exploration methodology that is defined to be the non-trivial extraction of implicit, previously unknown, potentially useful, 'relatively simple', and not predefined information from large databases (Fayyad et al. 1996). Of course, data must be sufficiently conclusive and contain relevant attributes for representative cases. Hence, KDD comprises a knowledge intensive task consisting of complex interactions, protracted over time, between a human and a (large) database, possibly supported by a heterogeneous suite of tools.

USE OF AN INTERESTINGNESS FUNCTION

An interestingness function formalises what is deemed to be important knowledge for the problem at hand. Importance should be interpreted in its widest connotation. To list just a few parameters that make up an interestingness function, one has:

- Novelty, which implies the (often unexpected) deviation from prior knowledge,
- Usefulness, which relates to the goal put forward (one is usually interested in a specific kind of knowledge)
- Simplicity, which refers to the syntactical complexity of the pattern
- Generality, which is determined by the fraction of the population the pattern refers to.

Hence, it is obvious that an interestingness function is to a large extent subjective. It determines which methods can

be used and which ones cannot. A pattern is then called knowledge if it is interesting enough, i.e., if the interestingness function trespasses a threshold.

THE KDD PROCESS

“Data mining” refers to a class of methods that are used in some of the steps comprising the overall KDD process. This term has mainly been used by statisticians, data analysts and the Management Information System (MIS) community. The term “KDD” has been used more by AI (Artificial Intelligence) and the ML (Machine Learning) community. Consequently, KDD systems typically draw upon methods, algorithms, and techniques from diverse fields such as ML, statistics, AI, exploratory data analysis, etc. Hence, it is common to find data mining methods from these fields. Unfortunately, terminology tends to be a bit confusing because it depends on the domain of application: e.g. dependent/independent variables, attributes/features, fields/records, test and validation set, A general definition of data mining and its link with KDD is given in (Fayyad et al. 1996) on page 9:

‘Data mining is a step in the KDD process consisting of particular data mining algorithms that, under some acceptable computational efficiency limitations, produces a particular enumeration of patterns over the data set.’

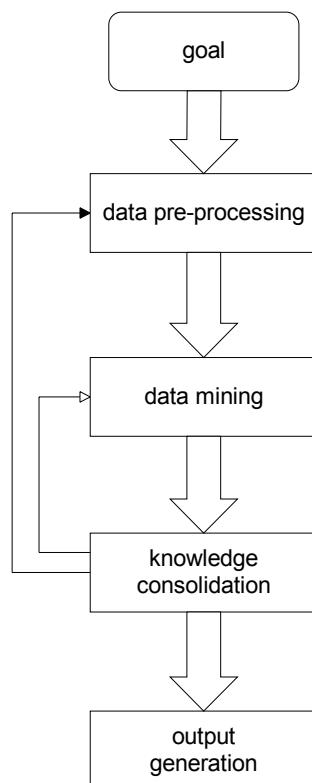


Figure 1: Steps in the KDD process

The basic steps in KDD are depicted in Figure 1.

One starts with the **goal definition**. For a goal put forward, one has to formulate the problem and get an understanding of the domain in which it is situated. A priori knowledge must be considered and used whenever possible. Some a priori knowledge may be present in a knowledge base. The goal should be formalized and made executable so that it can be related to relevant data that is hopefully present in the database. Example: identify the credibility of a potential customer for a mortgage loan.

The **data pre-processing step** consists of attribute focusing, data focusing, data cleaning, and (if necessary) data projection.

Attribute focusing involves an initial data reduction in the parameter space and it creates a target data set by focusing on a subset of variables that restrict the parameters used to do the analysis. Examples: possible relevant variables are occupation, marital status, possessions, bank accounts with amount of money in them, age, other loans, etc., while irrelevant variables are hobbies, religion, race, etc.

The data set may contain too many records and a second data reduction (in the observation space) may be obtained by looking at data samples that are relevant for the problem at hand. This is what is called data focusing. It is not to be confused with sampling the data for computational reasons. Example (ctd.): look at previously or existing customers that have a mortgage loan.

Data cleaning concerns the removal of noise, the analysis of outliers if appropriate, or the construction of a model to account for the noise. Additionally, a strategy for handling missing data has to be chosen. In data pre-processing all available a priori knowledge should be used (e.g., what about an outlier?). Data cleaning, however, is a double-edged sword: it is usually required to ‘clean’ poor data, but it may throw away a crucial indicator by considering it as an anomaly (e.g. outlier). Example (ctd.): outliers may be deceased customers, or cases where fraud is involved. In data projection, dimensionality reduction or transformation methods may further reduce the number of variables under consideration.

The **data-mining step** consists of a model specification, model fitting, model evaluation, and model refinement. Model specification involves deciding whether we use classification, regression, neural networks, or other techniques. The goal has a large impact: is comprehensibility more important (simple classification scheme) or is accuracy of prediction of major interest (recall the interestingness function)? Different techniques/models may be tried depending on the structure of the data. Some specific parameters of a model are determined based on a chosen training set. This is the process of model fitting. Model evaluation validates the patterns found on a simple test set or via cross-validation. Model refinement, depending on the evaluation results, iteratively refines the initial model or tries other model paradigms.

Consolidation of the newly found knowledge and output generation consist of interpreting the patterns found, and if more algorithms have been used, comparing them. Hence, it is decided whether the found patterns represent useful or interesting knowledge. Further consolidation of the discovered knowledge happens by summarizing and documenting it. Finally, the knowledge has to be integrated and conflicts (if any) that may arise with previous knowledge in the knowledge base must be resolved.

Looking at the previous steps, one notices that in the KDD process, data visualization is quite important (even in determining a goal; e.g., one notes something interesting from the data and one wants to investigate it further via a hypothesis test). Appropriate display of data points and their relationships (e.g., scatter plots, etc.) can give the analyst insights that are virtually impossible to grasp from looking at very large tables of data records (trends, dependencies, outliers, etc.).

KDD IN A BROADER PERSPECTIVE: THE VIRTUOUS CYCLE OF DATA MINING

In (Berry 1997) data mining (or KDD in our terminology) is itself a sub-process of what is called 'The virtuous cycle of data mining'. It goes on where KDD ends by transforming information into actionable processes. As Berry says: *'one must be able to respond to the patterns, to act on them, ultimately turning the data into information, the information into action, and the action into value'*. In other words, KDD itself is in fact only one major step in a even bigger business process that applies knowledge gained from increased understanding of customers, markets, products, and competitors to internal (continuously evolving) processes. The virtuous circle, which is depicted in Figure 2, is constructed by

- (1) identifying the business problem. This results in setting the goal in the KDD module.
- (2) using pre-processing and data mining techniques to transform the data into actionable information. This is the remainder of the KDD process module.
- (3) acting on the information. This action is not considered in the KDD.
- (4) measuring the results and putting them in the database for possible further processing. This action is not considered very explicitly in KDD, i.e., one may consider two kinds of feedback from the results to the goal.
 - (a) does it satisfy the goal?
 - (b) will the action satisfy the goal (is the action wrong, does the goal need adjustments)?Only item (a) is considered in KDD.

It can also be seen that item (1) and (3) are pure enterprise issues. These topics are the subject of management courses. Hence, the KDD approach delineated here com-

prises neither the business actions that have to be taken nor the consequences on new goal definitions.

The module 'measure results of action' refers specifically to measures of business value that go beyond response rates and costs, beyond average and standard deviations. It is more than just knowledge consolidation; it is looking at the real world effect of applying the information. Based on the results of the actions undertaken, one can define new goals or discover new business problems that have to be investigated further.

The virtuous cycle can also be applied to scientific discovery, where found information can often be tested by experiments (act on the information) and where as a result new goals (lines of research) can be set forward.

Figure 2 shows that a data warehouse delivers data to three processes: report generation, OLAP (On Line Analytical Processing), and KDD. The low-level data processing is done in a data warehouse or a data mart. The term data warehousing is very popular in MIS environments. It stands for collecting and cleaning transactional data and making them available for on-line retrieval. OLAP is a multi-dimensional data analysis that computes summaries and breakdowns along many dimensions. OLAP is superior to SQL (Structured Query Language) for analysis of data. Pure KDD goes further than OLAP. The latter only helps in answering specific questions (e.g., who are my best customers for a certain product?) in that it informs about general tendencies (patterns) in the data. KDD can better answer less trivial questions (e.g., what sort of customers should we target?).

To be able to link KDD with GST a brief overview of some relevant GST issues now follows.

GENERAL SYSTEM THEORY

Systems Theory concerns an interdisciplinary study that tries to find an abstraction that results in principles common to all complex entities, and the models that can be used to describe them. From a high abstractional level, there are two main routes that can be followed: the modelling (deductive) route and the identification (inductive) route. Of course, if a hybrid approach is possible it is advisable to try to use the much of possible of both alternatives.

The modelling route divides the system into sub-systems until one arrives at well-known sub-systems for which a model is available. The sub-models can then be joined together according to different methodologies to obtain a composite model, which represents the system under investigation. If the former approach is impossible, one has to treat the system as a black (or grey) box and try to infer a model via data-analysis of input and output signal recordings. This is the identification route, which is based on experimentation.

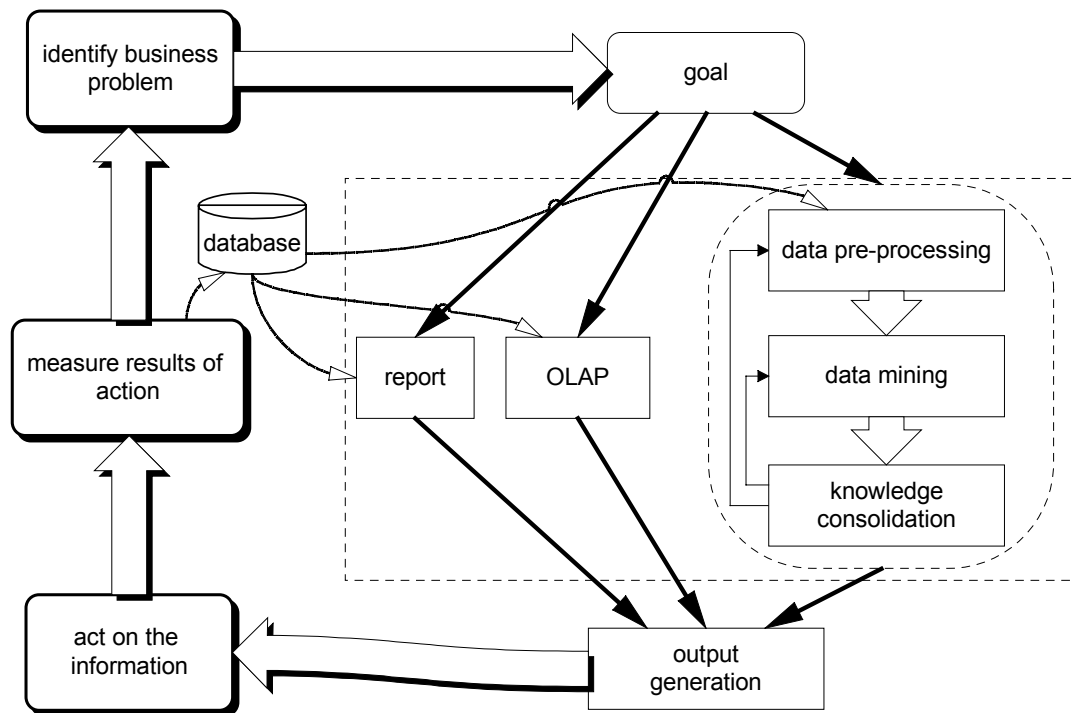


Figure 2 : Decision support and the role of data mining

The definition of a system is very abstract and thus applicable to every situation: 'a system is anything that is called a system'. Klir expresses general systems theory by saying: "A general system is a standard and interpretation-free system chosen to represent a class of systems equivalent (isomorphic) with respect to some relational aspects that are pragmatically relevant", (Klir 1985). From this, a taxonomy of general systems based on (abstract) relational aspects that transcends the disciplines of sciences can be constructed.

In this paper, elaboration is done on only one type of classification: based on the degree of deductive knowledge one has. White box systems can be completely modelled by use of a priori knowledge (well known inner structure and functioning), while black box is the other extreme: nothing is known about the system and one has to rely uniquely on observations. Hence, one has:

- knowledge and insight about the system (white box component)
- experimental data of system inputs and outputs (black box component).

The deductive approach is preferred whenever possible, (Karplus 1976). This is referred to as the physicality principle. Equations that follow from such physical principles are trustworthier than equations chosen on the basis of fit.

Depending on the amount of both sources of knowledge one has different shades of grey. Consequently, different approaches for investigating the system under consideration prevail.

A model is a surrogate for the real system under certain conditions (the conditions can be formalised in an experimental frame) such that, when solving a problem concerning the original system, it can solve the problem to better advantage. A model is a kind of system, so it too can be a black, white or grey-box. For example, general neural networks can be considered black box models, while tree classifiers are white-box models.

Model structure is a concept that may have different meanings:

- 1) the geometry and topology of the system, e.g., how the components are coupled in a lumped system, or the general configuration of a distributed system.
- 2) the type of the system description: e.g., differential equation form, etc.

In linking GST and KDD, validity concepts of a model are important. The notion of validity, formalized in (Coombs et al. 1954), is extended by Zeigler, who distinguishes different degrees of validity, (Zeigler 1976, p 5):

1. A model is *replicatively valid* if it matches data already acquired from the original system.
2. A model is *predictively valid* if it can match the data of the original system before these data are acquired from the original system. Predictive validity is stronger than replicative validity.
3. A model is *structurally valid* if it not only predictive valid, but also reflects the ways in which the original system operates to produce its behaviour.

Other ‘validity’ types will not be considered here (e.g. conditional validity: i.e. the model accepted as being conditionally valid until falsified).

Some directives should be taken into account when constructing or identifying a model. They reflect sound principles that stress the usefulness of the model. Firstly, a model must be able to explain empirical data in the sense that there must be close fit between the model output and the experimental data. However, the aim of science should be to find the simplest explanation; only if the simplest explanation proves unsatisfactory should the scientist move on to consider greater complexity. This corresponds with Occam’s razor principle, which can be found in the use of complexity measures, e.g. Akaike’s measure of fit. Additionally, complex models are difficult to calibrate and difficult to validate properly. Balanced accuracy comes into play when a global model consists of different model parts. Given a finite set of data and evidence, the global model will have its limitations and approximations, so it is wise to balance uncertainty over the different model parts. It makes little sense to strive for the most correct structure if that structure does not find the parameters or it does it only in a very poor way. A model is efficient if it is believed to be acceptable to a high degree if it is able to meet a variety of demands and if it is able to serve many purposes. It should at least serve the goal it is designed for.

In the last decades, a considerable amount of research has focused on model (and model structure in particular) characterization techniques. Each method developed consists essentially of a number of processing tasks:

1. conditioning the data to remove errors and noise ;
2. extracting a number of decision quantities, which summarize the information in the data set concerning the model structure;
3. deciding on a suitable structure utilizing the quantities just extracted

The major difference between the methods resides in the choice of the decision criteria.

Mathematical modelling has become a task which does not start from scratch every time the modelling activity is re-initiated, but is continued from the stored information obtained during previous runs. Hence, modelling often relies on stepwise-refinement methods.

Inductive Modelling And System Identification Methodology

System identification is the discipline of experimentally constructing mathematical models of systems, starting from observations on the system. Typically, a certain model structure with initially unknown parameters is chosen. Based on this structure, the parameters are identified. However, if the structure is not ‘right’, parameters are usually ‘tortured’ to make the fit work. Consequently, the model only represents the system well in a limited range.

The Concise Encyclopaedia of Modelling & Simulation describes identification as:

“Identification describes the search for a definition of a model showing the behaviour of a process evolving under given conditions. It is usually realized by means of a set of measurements and by observing the behaviour of a system during a given time interval. The model obtained by identification must enable the evolution of the identified process to be predicted for a given horizon, when the evolution of the inputs and various external influences acting on this system during this time interval are known”, (Atherton et al. 1992).

Hence, system identification is concerned with the problem of building mathematical models of dynamic systems based on I/O measurements.

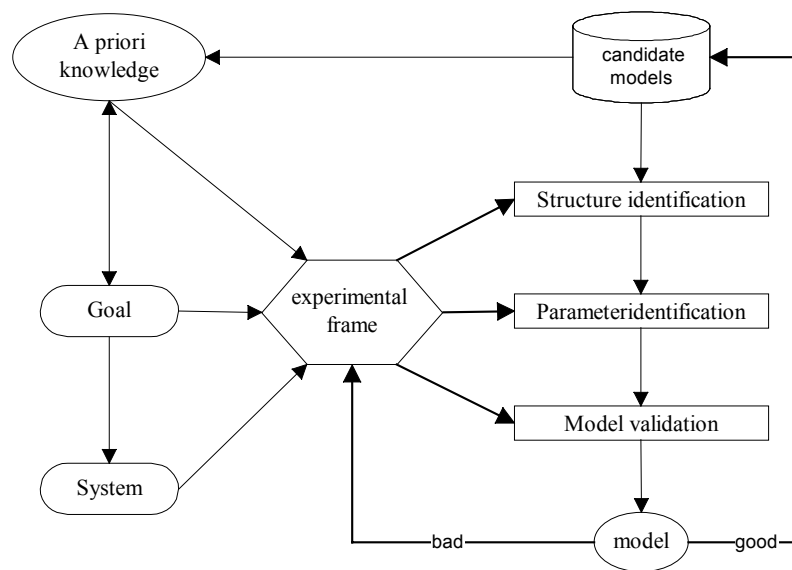


Figure 3 : System identification

The identification route is based on experimentation, but a mixed modelling-identification approach facilitates the maximum extraction of knowledge out of the system (if possible). The simplest approach is where deductive techniques result in an a priori model with known model structure. This would mean that in Figure 3 the structure identification is completely replaced by a deductive model construction. Parameter identification then delivers estimates for the yet unknown parameter values. The model validation encompasses in it also a validation of the given structure: if a good validation cannot be obtained, then probably something is wrong with the given model structure.

The approaches to identification as described in the Concise Encyclopaedia of Modelling & Simulation are quite general. They consist of

- collection of input-output data from the system; data are usually recorded by sampling in discrete time,
- settling for a set of candidate models,
- choosing one particular member of the model set as the best representative, guided by the information in the data.

The identification procedure consists of four choices:

1. Experiment design encompasses the choice of inputs to be made, sampling rates, pre-sampling filters, etc. to have the most informative data. Experiment design takes into account the goal, the a priori information, and the data (goal, a priori

knowledge and experimental frame blocks in Figure 3).

2. The choice of a model set is the most difficult one. A priori knowledge and engineering intuition or insight has to be combined with formal properties of model and identification methods to contribute to a good result. It is the structure identification block Figure 3.
3. The choice of criterion of fit concerns how to evaluate the quality of a particular model.
4. The model validation determines if a model is good enough, based on the criterion of fit.

PHILOSOPHICAL COMMONALITIES FOR GST AND KDD

In the sequel, a model is defined as any formalization that aids in understanding a system under investigation (conform the theory of general systems). Hence, a model in its most general form can be:

- a set of rules (if-then rules, association rules, ...) → both domains
- a set of differential equations, → GST
- a block diagram, → GST
- a Petri net, → GST
- a Bond graph, → GST
- a classification or regression tree, → KDD
- a neural net, → both domains

- a genetic algorithm, → both domains
- a dependency network, → KDD
- a hypothesis, → KDD
- a Markov chain, → both domains
- any qualitative model (e.g., confluences), → GST
- a look-up table or contingency table, → both ?
- a set of data (for lookup) → both ?
- a regression model (linear, non-linear, logistic, ...) → KDD
- a discriminant function, → KDD
- a clustering, → both ?
- etc.

The domain is suggested for which a model representation is well known, corrections, remarks, suggestions can be emailed to the author.

Anyhow, the list above is consistent with the general concept of a model: *"Something is a model of something else for an observer, if his/her relevant questions about the latter can be answered by the former"*.

In GST, either system observations may be obtained actively; one can generate the observations, or passively; observations are given as such. In KDD, the latter situation is the rule, while in GST both settings deserved equal attention.

The top-down approach to modelling is a deductive approach. It can be compared to hypothesis testing in KDD. In both cases, one has a priori knowledge that has to be cast in a formalized form, i.e., a model in its

most general meaning. A difference in the deductive approach is that general systems theory has a long tradition of handling systems and sub-systems (decomposition of systems and coupling of sub-systems), while KDD usually tackles only one (undividable) system.

The bottom-up approach to modelling is better known in general systems theory as system identification. Klir calls the bottom-up approach 'investigative', (Elzas 1984). The latter term pertains more to the term used in KDD, where the approach is known as data exploration. Both approaches are depicted in Figure 4.

In general, one can say the KDD is an inductive approach and thus well applicable to 'black box' systems. Its success is then also mainly in the corresponding domains of science, where little or no a priori knowledge is available (especially not in rigorous mathematical models such as ODE's). Hence, most applications of KDD lie in the field of economics, sociology, ecology and physiology. In the latter case, the model can be completely specified from a priori knowledge. Figure 5 shows the motivations for constructing a model and its use for different shades of grey, (Karplus 1976).

Inductive approaches are mainly found in the left hemisphere of the rainbow, while deductive approaches are found mostly in the right hemisphere. In some cases, one has some a priori knowledge about a system so the black box concept does not apply fully, but this knowledge is not enough to write down the model description in such detail (e.g. model equations). Hence, the situation for a 'grey box' problem is worse than a white box, but better than a black box. Thus, in the middle of the rainbow, mixed approaches are used. Note the shift from quantitative to qualitative when going more into the darker regions of the spectrum.

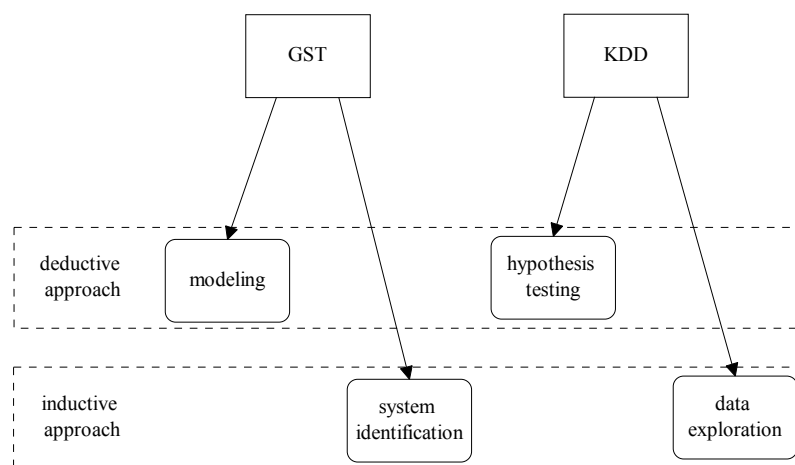


Figure 4 : Deductive and inductive approach in GST and KDD

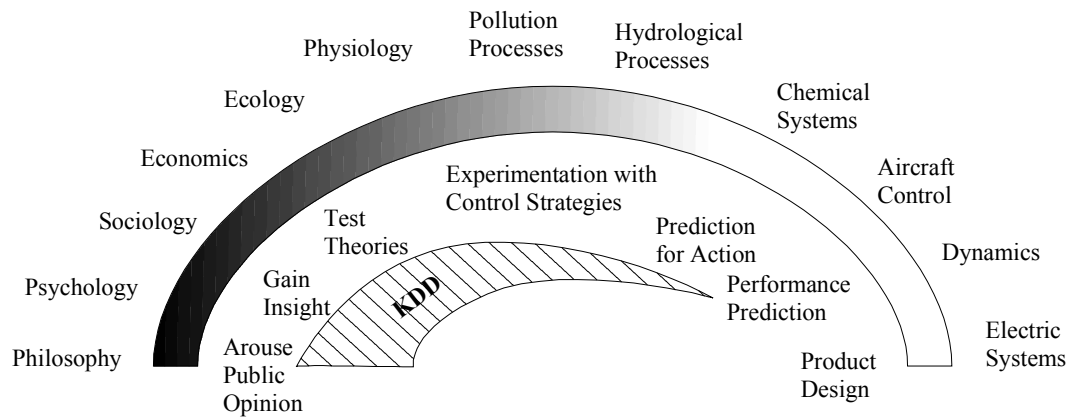


Figure 5 : Spectrum of modelling

With regard to the black box aspect, it is interesting to note the definition of a system according to Klir (which allows for both the deductive or inductive approach) where he says that it is sufficient to consider only the following constant traits of a system, (Klir 1969, p 52):

1. The set of external quantities together with the resolution level.
2. A given activity.
3. Permanent behaviour (e.g. a given relation between the quantities).
4. Real UC-structure (universal coupling)
5. Real ST-structure (state transition)

Each of the above traits can be used for a basic definition of a system. If the activity (2) is given (inductive approach or data exploration), we are interested in the other traits derivable from it:

- the set of quantities and the resolution level (1) follow directly from the form in which the activity is expressed, (GST and KDD)
- for a behaviour (3) it is relatively easy to verify whether or not it is consistent with the given activity, (GST and KDD)
- for a given set of state transitions (4), consistency can be checked, (GST and KDD)
- for a given universal coupling (5) it can be checked also (see Klir 1969 p 53), (GST)

Hence, a class of systems can be uniquely defined by given activity (series of I/O records). This corresponds with a system identification approach in GST, and with the starting point of data exploration in KDD. The first three points are commonly used in KDD, while the latter two (UC and ST structure) are more typical for GST.

THE LIFE CYCLE OF GST AND KDD

The similarity between general systems theory and KDD is proposed in a unifying context in Figure 6.

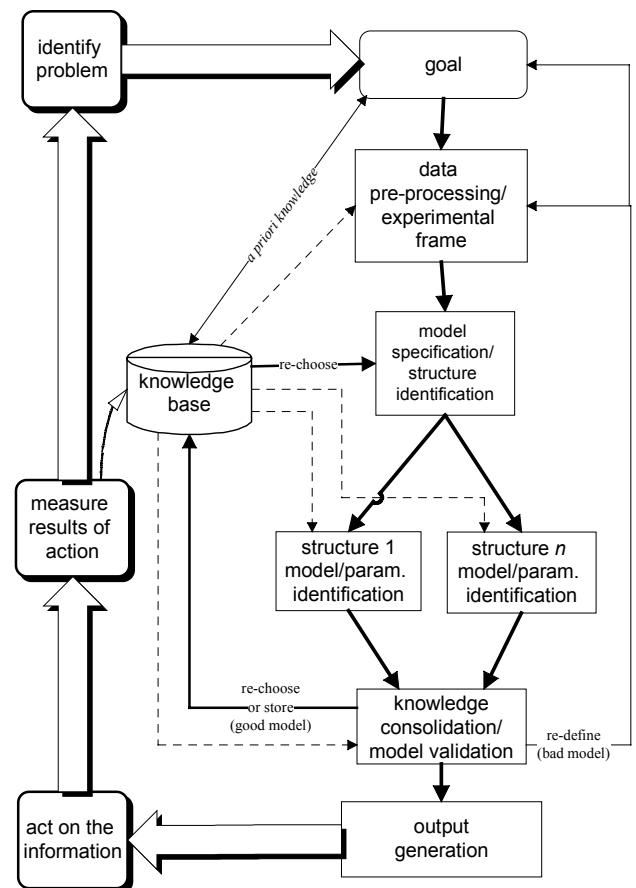


Figure 6 : General modelling and KDD

We now describe some important blocks of Figure 6.

Goal Formulation

All problem statements start with the identification of the problem and by trying to set a goal in order to solve the problem at hand. To be able to meet the goal it has to be formalized so that a rigorous approach to problem solving can be obtained.

In order to solve the problem and to satisfy the corresponding goal, a focus on part of reality has to occur: one has to define a system. Defining a system is studied in GST. It involves defining the boundaries of the system, the interaction of the system with the environment, and last but not least, defining the relevant variables (space-time resolution). Both KDD and GST stress the principle that a model should not be more complicated than absolutely necessary (Occam's razor). However, GST puts some more emphasis on a correct classification and prediction. Only the 'Quick decision' problem type, where accuracy may be less important than comprehensibility, is less stressed in classical modelling. The interestingness function defined earlier in KDD allows more flexible and general criteria settings to solve the problem at hand.

Three broad levels of intervention in GST that can be distinguished (management, control and design) can be found in the virtuous data cycle, which stems from the domain of KDD. Data mining can be used gaining insight (model purpose), and the found pattern in KDD will provide us with an opportunity to intervene at some level (as in modelling).

Data Pre-processing Or Experimental Frame Definition

Defining a system has much to do with the definition of the object system as it is defined by Klir, or with an experimental frame (as defined by Zeigler). An experimental frame isolates specific input/output behaviour both of the real system and its model, (Elzas 1984, p 44). In KDD, this corresponds with the data pre-processing step. Attribute focusing is in fact nothing more than defining what are supposed to be relevant variables for the system under investigation. This is consistent with the notion of an experimental frame and with what Klir calls an observation channel. It also fits the viewpoint taken in systematic modelling, which states that a model is conceived as a collection of variables and relations among them, (Ören 1984). The collection of relevant variables is the attribute focusing, while identifying the relations is part of the data-mining step. The KDD approach to attribute focusing is more related to the philosophical viewpoint of Klir than that of Zeigler. This is clear when considering what Elzas wrote: *'Imagine that we are able to prove that the model is not sufficiently detailed in its system description for some goal that was set beforehand. Confronted with this situation Klir will remark that the reason for this deficiency is a lack of knowledge, while Zeigler will consider the observation-frame to be insufficiently detailed'* (some text that was between brackets

or between '—' in Elzas has been deleted). Data focusing is less used in systems theory, because the nature of data is often different than for KDD. In KDD, one usually uses a collection of static independent records. Taking a relevant subset is the subject of sampling techniques and doesn't pose such a problem as in the case of dynamic data.

The field of sampling techniques is a field in its own right in KDD, while sampling techniques are known less well in GST. However, GST also considers other issues: e.g., data reduction can happen via the determination of the Nyquist frequency (the latter determines the time steps that are relevant for sampling). Again, it illustrates some of the different static(KDD)/dynamic(GST) modelling aspects of the respective domains. To the contrary, data projection can be applied in both domains. It can be meaningful to create a new variable, which summarizes the behaviour of a set of others, and to use that new variable in the modelling phase to be followed later. Data projection also fits in the definition of an experimental frame. Finally, data cleaning is appropriate whenever values are missing. This is especially true when considering system identification. For many problems in systems theory (think about controllers and other man-made devices), missing values are not of primary importance when compared to their role in KDD. Hence, modellers in the domain of systems theory rarely meet systems for which there are missing values (they do meet missing variables). Anyway, modellers in the latter domain can learn from the KDD domain about how to deal with missing values.

Another aspect is that the KDD society is used to handle all kind of variables (nominal, ordinal, continuous, etc). The GST society also knows about this taxonomy of variables, but they do not have such a systematic approach to constructing appropriate models for them. In the KDD society (especially statistics), one uses different models for nominal variables, ordinal variables and continuous variables (e.g., linear regression versus logistic regression, chi-squared based models on contingency tables versus other). Furthermore, the size of the data set is important. For small data sets, exact methods are used, while for larger data sets asymptotic methods can be used.

Model Specification Or Structure Identification

The data-mining step in KDD is strongly related to modelling in system theory. Model specification can be compared with model structure identification. It involves deciding what type of model we use. (in system theory: Bond graphs, Petri nets, block diagrams, ... in data mining: classification with decision trees, hierarchical clustering, linear regression, neural networks, etc.). The model specification step is straightforward, applicable to both domains and can be considered at different abstraction levels. On a very abstract level this involves choosing if one wants to use Bond graphs, Petri nets, block diagram, rules, neural nets, etc. (for

systems theory) or trees, clustering, rules, neural nets, regression models, etc. (for KDD). On a more concrete level, one has to elaborately specify the model. Examples are: linear, logistic, or non-linear regression (KDD), linear or non-linear models (GST), state-space or transfer functions in block diagrams (GST), kind of neural net (both domains), kind of tree (decision, regression or other kind of trees (KDD), order of a differential equation (GST), type of clustering (KDD), etc. In both domains, the goal has a large impact on the used model paradigms: when comprehensibility is more important certain representations may be more preferred than others. For example in GST, a block diagram may be more comprehensible than a Petri net (it is also relative to the field of expertise). A tree structure is more comprehensible than a neural net (KDD). Rules are more comprehensible than some other model paradigms (both domains), and general neural nets are usually the least comprehensible (both domains). The issue about comprehensibility has to do with the colour of the model: black box models are always less comprehensible than white box models.

Data Mining Or Parameter Estimation

Model fitting involves parameter estimation in modelling. It is also called model calibration, (Elzas 1984, p 51). In GST, this boils down to estimating parameters or coefficients in differential equations, the parameters in state-space models, etc. In KDD, similar estimation is done for regression models, tree models, etc. Neural nets parameters are the weights of the connections, etc.

Model evaluation is what is called model validation in systems theory. However, KDD goes further and compares totally different model structures (see knowledge consolidation). If we stick to the 'classical' model validation, then comparing the behavioural data of the system under investigation and the calibrated (fitted) model is mostly done via a train-and test paradigm. For large data sets, a rule of thumb is to use 2/3 for model fitting (training) and 1/3 for validation (testing). Cross-validation is commonly used in KDD, while it is less popular in GST. Even more specifically, bootstrapping is known too in KDD, but almost unknown in GST. Replicatively validity is known in GST: it consists of fitting the model on the training set. In KDD, this is better known as internal estimates (resubstitution estimates, [Breiman 1984]). Predictively validity is done on a test set; it gives the real estimates. What Elzas calls 'realism', i.e., looking at structural isomorphisms at different degrees of lumping of sub-systems (and corresponding sub-models), is not considered in KDD. There is usually only one global level, and the validation takes place for the total system (no UC structure).

KDD uses an interesting function for evaluating a model. The concept of an interesting function is more general than a simple evaluation function that is usually based on accuracy performance. Hence, KDD takes more into account that is economically interest-

ing. This may prove an important point for the modelling society when they want to evaluate their models in an economic context or when speed is of the utmost importance. KDD provides a better framework for dealing with these situations.

Model refinement is equally applied in system theory and KDD; when a model does not validate well, another model (structure) is chosen. When this fails too, one can go one step further back and redefine the experimental frame or adjust the goal. The refinement of an existing model is a big issue in modelling. The systems theory as developed by Zeigler (use of a SES (System Entity Structure) and model base) is designed to construct and refine models.

Principles of good architecture such as consistency, orthogonality, property, parsimony, transparency, generality, open-endedness, and completeness, are equally applicable in KDD. This is because a model is a very abstract concept, which is so general that it encompasses the patterns that are sought and the hypothesis that may be tested in a KDD environment.

From Figure 6 it can be seen that many models may be used in parallel. The models can be of a different paradigm (e.g., neural nets and genetic algorithms) and they can contain in addition other 'deeper' levels of knowledge (e.g., rules versus decomposed models). In the latter case, one speaks of shallow or deep models (in GST). The concept of shallow and deep models is also found in the KDD environment: report generation is shallower than OLAP, and OLAP is shallower than KDD. In KDD, much attention is paid to comprehensibility of models, so a rule model can be investigated next to a tree classifier. They do not differ much in the expressiveness, but the machine learning society does make a difference with regard to how easily they can be understood.

Knowledge Consolidation

The knowledge consolidation step compares the different models used. Here, model evaluation is on another epistemological level than in the data-mining step. Models are not only compared on a validation set to validate the parameter estimation, but they are compared on a test set to 'validate' or test the chosen model paradigm. This is clearly an abstraction level higher. The GST community has not done so much work in this aspect as the KDD community that focused and implemented this right away. Model switching is used in GST, but there the purpose is more to see if a model is still valid for time-dependent behaviour. Hence, the underlying paradigm is different; they are orthogonal: one deals with comparing models for the same behaviour, the other deals with models for changing behaviour. Luckily, they can both be used when studying time-variant behaviour. One can switch from a state-space model to a neural net when the behaviour goes outside the specifications of the state-space model (i.e., when another structure is more appropriate). It could

even be that, if the system goes out of the preset specifications (and thus the state-space model fails), the goal settings change accordingly (e.g., speed becomes more important than accuracy). In GST it is known that, for example, linear models may have to be replaced by non-linear models in certain circumstances, but this is more because of the fact that linear models are often an approximation of non-linear systems, an approximation only valid within certain working parameters (experimental frame).

In GST, the newly found model can be stored in a model base, while in KDD this is not stated so explicitly. In the latter case, it is sometimes stored as previous strategic info. Hence, there is room for elaboration from both sides.

Usually, output generation is heavily based on appropriate graphical representations of the data. From that point of view, GST can learn from KDD, because the latter is well acquainted with all kinds of graphs (e.g., multidimensional graphs such as Trellis graphs).

The rest of the picture (acting on the information, measuring the results of the action and storing them, perhaps recognizing another problem type) is so general that is valid for any modelling attempt, be it via GST or via KDD.

Remark

In Figure 6 the knowledge base is not split up in a model base and a database of records because a matrix of records can be considered a model too. Rules that may serve as a priori knowledge (or even written information in plain English) can be considered as models too, be it of a less formalized level. Therefore, a general term could be 'model base', but it is more appropriate to make it even more general by using the term 'knowledge base'.

CONCLUSIONS

The tendency in GST is more on the dynamic structure of models, while KDD is more focused (until now) on the static structure of models.

It is demonstrated that modelling in its most general meaning consists of the same steps in both the domains of general systems theory and knowledge discovery in databases. It has been shown that there are some paradigm shifts, but nothing prevents us from merging them to make them available to both domains. KDD focuses a bit more on the comprehensibility and interestingness of a model, hence making it more general with regard to a goal definition.

GST focuses more on time-dependent data and on model decomposition. For complex problems, the decomposition aspect can be used in KDD too. It may be better to split up a very complex system into subsystems and to tackle each subsystem in turn and then

to cast the found information in a framework that keeps count of the coupling between the sub-systems.

Finally, KDD is mainly based on an abstraction approach, while GST also included from the beginning a hierarchical approach.

A merging of both domains seems feasible and the resulting meta-modelling methodology seems to be beneficial for both domains.

REFERENCES

- Atherton et al. 1992. Concise Encyclopaedia of Modeling & Simulation, Ed. D.P. Atherton, P. Borne, Pergamon Press.
- Berry M. J. A., Linoff G. 1997. "Data Mining Techniques For Marketing, Sales and Customer Support", Wiley Computer Publishing.
- Breiman L., Friedman J. H., Olshen R. A., Stone C. J. 1984. Classification and Regression Trees, Chapman & Hall.
- Coombs C.H., Raiffa H., Thrall R.M. 1954. "Some Views on Mathematical Models and Measurement Theory", Decision Processes, John Wiley, New York.
- Elzas M.S. 1984. "System Paradigms as Reality Mappings", in: *Simulation and Model-Based Methodologies: An Integrative View*, ed. Ören T.I., Zeigler B.P. and Elzas M.S., NATO ASI Series, Series F: Computer and System Sciences, vol. 10, Springer Verlag, p 41- 68.
- Fayyad U. M., Piatetsky-Shapiro G., Smyth P. 1996. "From Data Mining to Knowledge Discovery: An Overview" in: *Advances in Knowledge Discovery and Data Mining*, ed. Fayyad U. M., Piatetsky-Shapiro G., Smyth P. and Uthurusamy R., AAAI Press/MIT Press, Cambridge, England, p 1 - 34.
- Karplus W.J. 1976. "The Spectrum of Mathematical Modeling and Systems Simulation", *Simulation of Systems*, ed. Dekker L., North-Holland Publishing Company, p 5 - 13.
- Klir G.J. 1969. An Approach to General System Theory, Van Nostrand Reinhold.
- Klir G.J. 1985, Architecture of Systems Problem Solving, Plenum Press.
- Ören T.I. 1984. "Model-Based Activities: A Paradigm Shift", in: *Simulation and Model-Based Methodologies: An Integrative View*, ed. Ören T.I., Zeigler B.P. and Elzas M.S., NATO ASI Series, Series F: Computer and System Sciences, vol. 10, Springer Verlag, p 3- 40.
- Zeigler B.P. 1976. Theory of Modelling and Simulation, John Wiley & Sons.

The author gratefully acknowledges support received from the EU IST Programme's sponsorship of the EUROSIGNAL project.

SIMULATION METHODOLOGY

Simulation Languages and Tools

First-Year Science and Engineering Students Can Simulate

John Miller, Professor of Physics,
Champlain Regional College,
Campus Champlain-St.Lawrence
790, Nérée-Tremblay
Sainte-Foy, Québec, Canada G1V4K2
E-mail: john.miller@utoronto.ca

KEYWORDS: elementary simulation, college-level, Maple V, scientific method

ABSTRACT

This paper summarizes the approach used at Champlain Regional College (St.Lawrence Campus) to introduce Physics students (pre-engineering) to computer-based modeling of physical systems. Maple V is the language used. Simulation is integrated into the scientific method followed by the students in the laboratory.

INTRODUCTION

Questions

Simulation is currently used as a pedagogical tool in many parts of the Science and Engineering curricula in a variety of educational institutions. Why should it be taught? When should it be taught? What and how should be taught? Tentative elements of answers are suggested in the discussion below.

Requests

We solicit input (criticism, suggestions) from industrial and academic designers and users of computer models.. We would also like to request and encourage those who practise computer-based modeling to become partners in local educational institutions as mentors, both for the professors and for the students.

WHY STUDENTS NEED TO SIMULATE

Learning to think scientifically

One of the general objectives of a Science program is to teach students to think scientifically. The scientist (whether pure or applied) explains data by creating an imaginary structure, a scientific model, which "behaves like" the real physical system. An analysis of the behaviour of the model enables the scientist or engineer to predict other behaviour of the system which until now may not have been observed. An experiment is then done to verify those predictions and hence verify the model. "If we do X to the physical system it will do Y" if our model is valid. If we in fact observe Y when we do X then that validates the model.. If the model has been confirmed often (i.e. one has confidence in the model as being functionally isomorphic with the real

system) the applied scientist or engineer can use the model to develop new applications.

Why simulation is necessary

For the simplest of physical systems it is possible to analyse the behaviour of the systems analytically, by the solution of simple algebraic and differential equations. The hard reality is that the thousands of problems at the end of the chapters of many textbooks describe unrealistically simplified systems. The equations that come out of scientific models which represent real systems of any complexity usually cannot be solved analytically. Numeric techniques must be used. Dynamic simulation is an effective way to find the implications of the scientific models.

The Black Box of Commercial Simulations

Students use many computer simulations for pedagogical and entertainment purposes. The simulations may be commercially programmed or programmed by local faculty. In general, however, these simulations are "black boxes". Students are not equipped to verify the validity of such simulations. They are "blind consumers".

List of Result of Learning Simulation Techniques

Students who program their own simulations, even for simple physical systems...

- i. develop a deeper understanding of the scientific models they are learning
- ii. have a better understanding of the pathway process of solving problems.
- iii. learn to appreciate the skill of professional programmers
- iv. may consider computer science as a possible career choice...or not!
- v. acquire confidence in their own ability to program simulations for their own needs OR recognize when to call on somebody with greater competency

WHEN SHOULD SIMULATION BE TAUGHT?

Our Students

"College" ("Cégep") in Québec province of Canada refers to the first level of post-secondary education in which the focus is both on "general" courses (language, humanities, physical education) and on specialized pre-university

courses., College-level students in Pure and Applied Sciences or in Health Sciences (leading to studies in the sciences or in engineering) have completed one or two algebra-based Physics courses at the secondary school level and are now somewhere in a sequence of three compulsory Calculus-based courses in mainly classical Physics. They also follow a two-semester Calculus sequence and a first course in linear algebra. Although we find that more and more students arrive at the college level with some programming experience, in general our students have not studied programming and are not taking programming courses.

We have found that the mathematical level of these students and the level of their Physics courses provides a satisfactory basis for introducing the concepts and techniques of basic simulation.

HOW SHOULD SIMULATION BE TAUGHT?

The Language Used

We have developed local simulation development environments (SIDE) in TurboPascal, in Java, in Java applets, and in JavaScript. We are currently beginning a similar project in Visual Basic. Although these environments are useful, they hide a lot of the programming from the student. Students merely entered the "core algorithms" describing the physical system under study. Much of the "dirty work" has already been done. We wanted to give our students the experience of programming the models themselves, from the ground up.

Our current generation of students uses Maple V, because it is available on our computer system and because we have found that Maple V is user friendly for beginning students. Some students are familiar with

Maple V is a comprehensive computer language using mainly symbolic computation, but also with numerical capabilities (Monagan et al, Learning Guide, 1998). It has good graphing capabilities to represent the results of running a simulation.

Since the students are learning programming by doing the simulation exercises we are limited to basic syntax. The following list will give the Maple V programmer an idea of what we expect students to learn beyond basic mathematical operations:

if..then,for..from.,arrays, procedures

Note that, as is the case of all programming environments, there are some frustrations for the student as they program in Maple V, such as incomprehensible error messages!!

Object-Oriented Programming? We try...

In the early stages of teaching modelling and simulation we use a simplified form of object-oriented programming.

Even if the programming language we are using (see the next section, below) is not object-oriented, we insist that students think in this mode for two reasons. First, object-oriented models are more isomorphic to the object structure of the real world and hence are more intuitive. Secondly, it will prepare them for object-oriented programming.

Students must identify the interacting objects in the real physical system then program them in Maple V. They are asked to make the computer "think it is the physical system under study."

Students learn that real objects can be represented by Maple V procedures. Real objects have properties which are represented by the values given to variables in the procedure. Real objects respond to other real objects (e.g. a falling ball responds with an upwards acceleration when it "detects" that its position is the same as the position of the ground. This is effected by "methods" (procedures or other code sequences) in the simulation object.

The idea of "class" is introduced in our pedagogical approach by explaining that the general plan of an object, before values are assigned to the specific object, is a "class". Instantiation is done manually by the student assigning values, in the simulation program, to the property variables.

Multiple objects from the same class may be distinguished by using arrays of properties. Object 1 uses the first element of the mass array, for example. In this first introduction to simulation this concept was rarely used except in optional Physics courses in the context of research projects..

We do not even refer to "advanced" object-oriented concepts such as inheritance.

The Importance of doing experiments

In our context, simulation is part of the scientific approach.. The computers follows the logic inherent in the scientific models underlying real systems to make predictions concerning the behaviour of those systems. Students then test those predictions by experiment and draw appropriate conclusions

EXAMPLE OF A "FINAL PRODUCT"

The simplest simulation seen by students is the "falling rock" (vertical projectile). This example is used to introduce the students both to the Maple V language and to the concept of computer simulation. Below is one student's successful attempt to program the falling rock. At the end the program calls up the following procedures:

- 1-Initialization of the simulation
- 2-Initialization of the system
- 3-The simulation itself
- 4-Graphical output

>#--The Falling Rock--

># 1-Initialize the Simulation

```
>siminit:=proc()
> global t,tt,ndp,dt:
> # t=time tt=total time
> # ndp=number of data points
> # dt=time increment between points
> local i:
> ndp:=1000:
> t[1]:=0:
> tt:=3:
> dt:=evalf(tt/(ndp-1)): #ndp-1 time increments
> end:
```

>#2-Initialize the System Model

```
> rockinit:=proc()
> global y,vy,ay:
> # y is the position of the rock
> # vy = velocity, ay= acceleration
> y[1]:=0:# Initial y
> vy:= 10:# Initial vy
> ay:= -9.8: #Acceleration
> end:
```

>#3-Let the Model "Run"

```
> rockfall:=proc()
> global t,tt,ndp,dt,y,vy,ay:
> local i:
> for i from 2 to ndp do
>   t[i]:=evalf(t[i-1]+dt):
>   # evalf ensures that the result is real
>   vy:=evalf(vy+ay*dt):
>   y[i]:=evalf(y[i-1]+vy*dt):
> od:
> end:
```

> #4-Graphical Output

```
> graphdata:=proc()
> global t,tt,ndp,dt,y,vy,ay,dp:
> local i:
> dp:=[seq([t[i],y[i]],i=1..ndp)]:
> plot(dp); #Graph of y=f(t)
> end:
```

> #5-Procedure Calls

```
> siminit():
> rockinit():
> rockfall():
> graphdata():
> #End
```

This obviously could have been programmed without procedures. Insisting on procedures is an attempt, albeit weak, to keep the student thinking "object-orientedly".

Other Student Simulation Projects

Some other topics where we integrate simulation into our pedagogical approach are the following: *projectiles bouncing balls, charging and discharging capacitors, oscillations, interference of waves*. In the context of "Physics research projects" or "Engineering application projects" students sometimes have done much more complex simulations.

CONCLUSIONS

Our experience has demonstrated that beginning students can program computer models of simple physical systems. It is gratifying to see students who say "I hate computers" finally come to the point where they are excited when they finally see the "bouncing ball" graphically, on the screen.

Our most exciting experiences have been when, in the context of laboratory research projects, students were paired with experts in the field. This mentoring relationship is used only for "research projects" undertaken in advanced Physics courses.

Based on that experience, we would like to encourage industrial and academic specialists to make themselves available for such consultation. Both the mentor and the student will find this experience enriching.

The preliminary ideas outlined in this paper will be the starting point for more formal research in this area.

REFERENCES

We have found the following books useful as a reference for teachers who are learning about simulation programming:

Giordano, N.J. *Computational Physics*. Prentice Hall 1997
Gould, Harvey and J.Tobochnik *An Introduction to Computer Simulation* Part 2) Addison-Wesley, 1988

The following references are made available to students for learning Maple V syntax:

Monagan,	M.B.	et	al.	Maple	V
Release 5: Learning Guide. Springer 1998-1					
Monagan,	M.B.	et	al.	Maple	V
Release 5: Programming Guide. Springer 1998-2					

BIOGRAPHY

JOHN MILLER was born in Buenos Aires, He spent most of his life in Canada.. His undergraduate studies were in Biochemistry and his graduate studies were in Biophysics, both at the University of Toronto, Canada. For the last 27 years he has been professor of Physics at Champlain Regional College. One year from retirement?

TAPE PERFORMANCE MEASURES IMPLEMENTED IN AN INCIDENT MANAGEMENT MODEL

R. Van Loock
F. Van Utterbeeck
H. Pastijn
Royal Military Academy
Renaissance Avenue, 30
B-1000 Brussels
Belgium
E-mail: Raf.Van.Loock@rma.ac.be

KEYWORDS

Discrete event simulation, business process re-engineering, performance measures.

ABSTRACT

Various consultancy companies propose the use of the so-called TAPE (Timeliness, Accuracy, Productivity, Efficiency) performance measures to assist business process re-engineering. We study their implementation and their relevance in comparison with traditional performance measures. We illustrate this by means of an incident management model for a call centre.

INTRODUCTION

The primary problem when comparing alternative designs of a system is to determine which performance measures to use. In a manufacturing environment, popular performance measures include throughput, lead-time, waiting time in queues, delays, etc. These measures are well described in the literature (Law and Kelton 1991). For business process re-engineering, various consultancy companies propose other performance measures, including the TAPE performance measures, that can be used as ranking criteria when comparing the alternative designs of a system. Most of the time these are used for system analysis by hand. We investigate their relevance for the comparison of alternative system designs by means of computer simulation.

First we will describe a simulation model of a call centre which was implemented in ARENA[®]. Then we will discuss the four TAPE performance indicators and describe how to integrate them into the simulation model. We will illustrate their use with results obtained from the simulation of the call centre model. Finally we compare alternative designs of this model based on the TAPE performance indicators and based on the traditional performance measures like waiting times in queues, resource parameters and service level.

INCIDENT MANAGEMENT MODEL FOR A CALL CENTRE

The simulation model we consider is the incident management process of a call centre. We will describe the elements of the model (the incidents, the resources and the skills matrix) and the process flow. The process flow is illustrated in figure 1.

The *incidents* are initiated by the customers of the call center. These incidents are represented by the calls that arrive at the centre. These *incoming calls* follow a stochastic arrival pattern. The calls are subdivided into 4 categories and 33 subcategories, depending on the area of expertise required by the customer. Each category has a specified probability of occurrence, while the subcategories within a certain category are assumed to be equiprobable.

The *resources* in our model are the dispatcher(s) and the system engineers. Each resource has its own weekly working schedule and has a FIFO queue associated with it. Incoming calls will wait in the FIFO queue if the resource is busy. A call will be *rejected* (and leaves the system immediately) if the time spent waiting in a FIFO queue exceeds a certain fixed threshold.

Every system engineer has his own areas of expertise, which are specified in the *skills matrix*. Every line in the matrix represents a subcategory, while every column represents a system engineer.

The *process flow* used in our model can be summarized as follows. Every incoming call must pass through a dispatcher. The dispatcher will rout the call to a system engineer whose area of expertise covers the category and subcategory of the call. If multiple system engineers are eligible, the dispatcher will rout the call to the resource with the shortest queue. Ties are broken in favour of the resource located the most to the left in the skills matrix. The *dispatching time* (the time needed by the dispatcher to decide on the routing of the call) follows a triangular distribution.

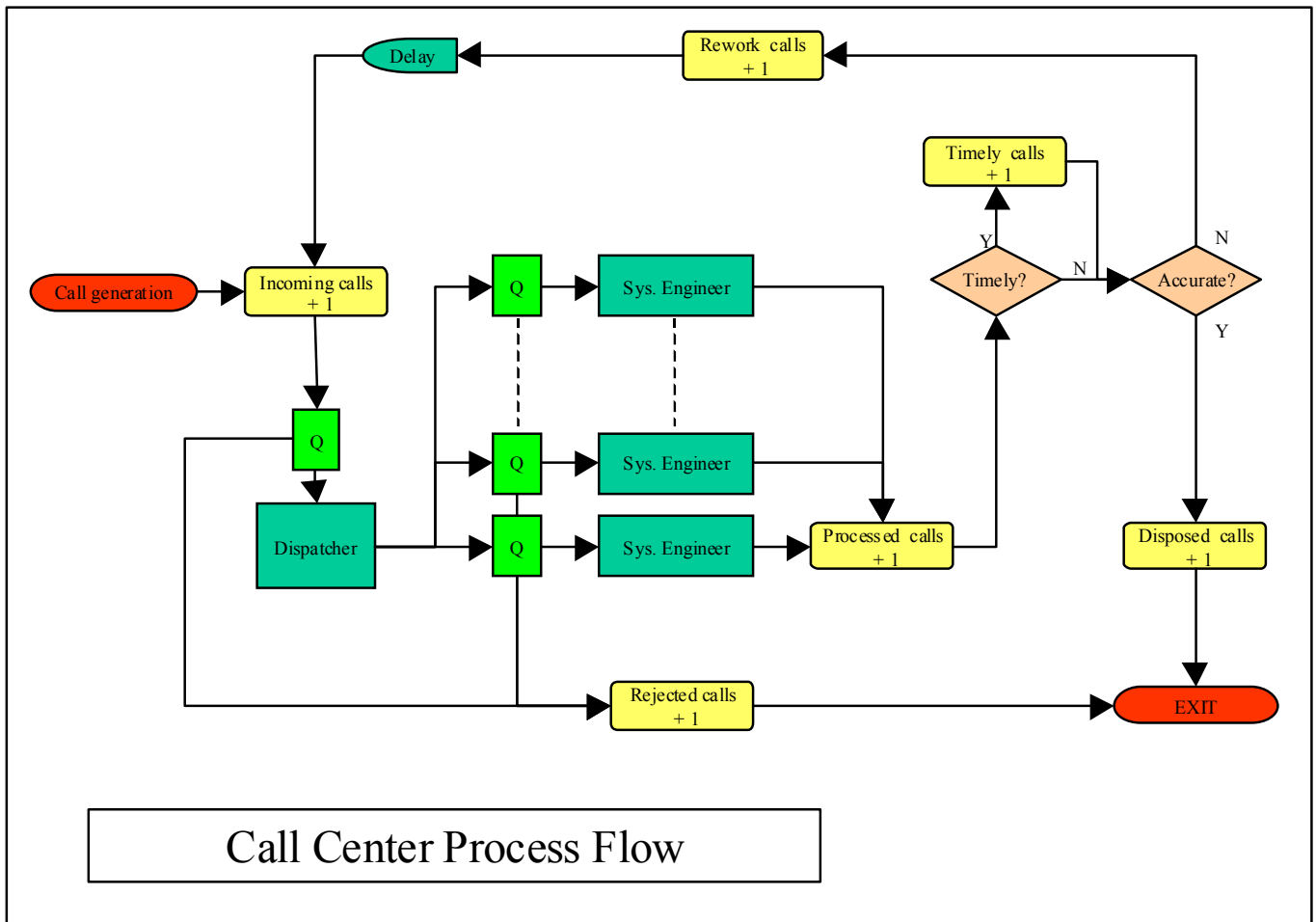


Figure 1

The *processing time* (the time the system engineer needs to handle a call) is assumed to follow a specified triangular distribution, regardless of the subcategory. For every *processed call*, there is a fixed probability that the customer is not completely satisfied with the assistance provided. These customers will call back after a stochastic delay. These subsequent *rework calls* will result in a decrease in the performance of the call centre. If the customer is satisfied with the assistance provided, the call is *disposed* and leaves the system.

PERFORMANCE INDICATORS

The four so-called TAPE performance indicators, which are proposed by some consultancy companies, are timeliness, accuracy, productivity and efficiency. We investigate how to integrate these indicators into the computer simulation model described above.

Timeliness

We define the timeliness of the system as the percentage of the incoming calls that are considered *timely calls*. A call is considered timely if the time spent in the system until it has

been processed by a system engineer does not exceed the *timeliness threshold*. The timeliness threshold is considered as an input parameter of the model. Note that rejected calls can never be timely.

Accuracy

The percentage of rework is determined by the accuracy of the system. While it is possible to measure this on a real-life system, it has to be implemented as an input parameter for computer simulation models. Rework calls will be initiated when a customer is not satisfied by the assistance provided by the system engineer. The accuracy will be considered as the input parameter of the system which defines the probability that a processed call will result in a rework call.

Productivity

We define the productivity of the system as the average productivity of all the resources. Note that this may be a weighted average if the working schedules of the resources are not balanced. The productivity of a resource is the percentage of the working time of the resource during which the resource is not idle. We suppose that a resource

is “adding value” to the system when busy. This indicator corresponds with the so-called scheduled utilisation of a resource which is a standard measure in ARENA®.

Efficiency

A lack of accuracy within the system will cause rework calls, as described above. The subsequent call will require the same area of expertise as the original call, and the resulting rework required of the available resources will decrease the efficiency of the system. The efficiency of the system is defined as being the average efficiency of all disposed calls. The efficiency of a disposed call is defined as the *total handling time* (dispatching time + processing time) of the initial call divided by the sum of the total handling times of the initial call and all subsequent rework calls. For example: the efficiency for a call without rework is 1, the efficiency for a call followed by two call backs (assuming an equal total handling time duration for all three calls) is 0.33.

SIMULATION

Several different configurations of the system were simulated, and the observed performance indicators were compared. Reconfiguration of the system is simulated through changes in either the number of resources (additional system engineers or additional dispatchers), or through changes in the skills matrix. More radical changes, like the implementation of a frontoffice-backoffice strategy, were not considered.

The screenshots in figure 2 and 3 give some examples of the output provided by the simulation. Each run simulated 10000 minutes (nearly one week) of call center operation.

Simulation Results				
Simulation time:	10000,00 min			
Timeliness:	74,55 %			
Productivity:	57,59 %			
Efficiency:	95,80 %			
Number of sys. engineers:	12	Incoming calls:	7578	
Number of categories:	4	Rejected calls:	312	
Number of subcategories:	33	Processed calls:	7200	
Service Level	95.88 %	Rework calls:	706	
		Disposed calls:	6872	
		Timely calls:	5417	
Resource name	Number of calls seized	Scheduled utilization	Average queue length	Average queue waiting time
VG	895	0,85	0,12	1,30
LD	938	0,88	0,13	1,32
AK	546	0,53	0,03	0,62
JP	598	0,57	0,04	0,70
LVC	735	0,70	0,05	0,74
WW	621	0,59	0,04	0,68
TL	476	0,45	0,03	0,60
FS	494	0,47	0,03	0,59
JC	520	0,50	0,03	0,66
NDM	754	0,71	0,07	0,94
WC	411	0,39	0,02	0,54
PV	278	0,26	0,01	0,34

Figure 2

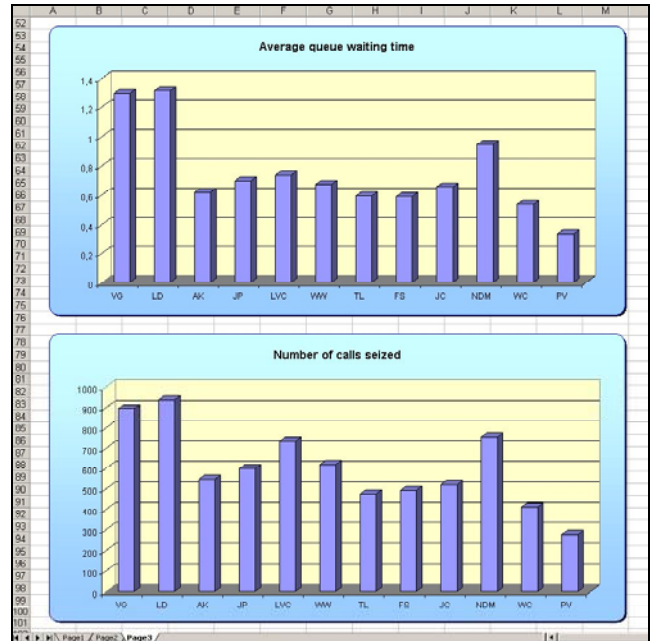


Figure 3

We illustrate the relevance of the 3 implemented TAPE performance indicators with the following simple scenario: in phase 1 we consider an overstaffed call centre (15 system engineers (SE's)) that relies heavily on a few “key players” (SE's with a very broad area of expertise), while several other SE's are “newbies” (with a very limited area of expertise). In phase 2 we simply eliminate the three least productive SE's. At this moment, the skills matrix still contains 12 subcategories which are only covered by a single SE. In phase 3 we start cross-training our personnel: the 3 least-skilled SE's each receive training in 2 subcategories which were previously only known by a single SE's. Finally, in phase 4, we finalise the cross-training effort in such a way that every single subcategory is now covered by at least 2 SE's.

Figure 4 illustrates the evolution of the timeliness, productivity and efficiency of the system over the 4 phases of our scenario. These results were based on 10 replications, resulting in 95%-confidence intervals with a width of less than 2%.

Both timeliness and productivity provide a quick and easy assesment of the “quality” of a certain system configuration. In phase 1 of the scenario described above, the low productivity indicates that the number of resources is more than sufficient, while the low timeliness indicates that there is a shortage of certain skills, causing long queues in front of key personnel. The combination of both indicators strongly suggests that more crosstraining could lead to a better configuration (requiring less personnel). As expected, the elimination of the least productive resources in phase 2 results in a greatly improved productivity, while the impact on the timeliness remains relatively limited. The first crosstraining session produces the expected results: both productivity and timeliness improve. The final

crosstraining session has no noticeable impact on timeliness, but still results in a minor increase in productivity. The high and relatively stable productivity level at this stage seems to suggest that more well-trained personnel will be necessary if we want to improve the timeliness. The impact of the considered system reconfigurations on efficiency is negligible, mostly due to the simplicity of the model and the changes considered. A more radical change (like the frontoffice-backoffice strategy mentioned earlier), is required before this indicator becomes relevant.

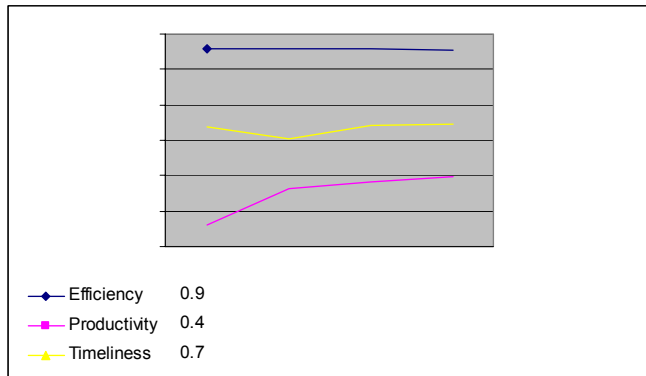


Figure 4

While the timeliness and productivity provide a good assesment of the overall quality of a configuration, one easily understands that more detailed indicators, like the traditional queue lengths, are essential for an in-depth understanding of the dynamics behind the model. It is for instance impossible to base the decisions that led from phase 1 to phase 4 in the example above (which SE's to eliminate, how to organise cross-training) on the 3 implemented TAPE indicators alone.

CONCLUSIONS

We have shown how to integrate three of the four TAPE performance indicators (timeliness, productivity and efficiency) in a computer simulation model of incident management processes for a call center. The fourth performance indicator (accuracy) had to be defined as an input parameter. These concepts can easily be applied to most other models.

Simulation has shown that the top-level indications provided by these three performance indicators can be used as a simple and relatively powerful tool during the first phase of a business process re-engineering project, where one wants to compare fundamentally different designs of a system. Present research however has revealed the limited sensitivity of these TAPE performance measures to smaller changes in the system design, which suggests their relative irrelevance during the final fine-tuning phases of business process re-engineering projects. The traditional

performance measures remain the performance indicators of choice for more refined and detailed analyses.

REFERENCES

- Kelton, W.D., Sadowsky, R.P., and Sadowsky, D.A. 1998. *Simulation with Arena*, WCB/McGraw-Hill.
- Law, A.M. and Kelton, W.D. 1991. *Simulation Modeling and Analysis*. 2nd ed, McGraw-Hill, New York,
- Van Loock, R. and Pastijn, H. 2003. *Performance measures in simulating incident management processes of a call centre*. Orbel 17, Brussels Jan 23-24 2003

Adding instrumentation tools to the SMPL discrete-event simulation language*

X. Molero, V. Santonja, J.A. Alegre and I. Torregrosa
Departament d'Informàtica de Sistemes i Computadors
Universitat Politècnica de València
Camí de Vera, s/n. 46022 València (Spain)
e-mail: xmolero@disca.upv.es

Keywords: discrete simulation, model instrumentation, SMPL and CSIM simulation languages

Abstract

SMPL is a discrete-event simulation language used at the universities of many countries in order to teach simulation. This language is appropriate for beginners due to its simplicity and easy use. Until now, we have not found in the literature significant improvements to its implementation. In this work we have improved the modeling capabilities of SMPL by adding several instrumentation tools. They can be used to collect simulation variables such as queueing times, queueing lengths, or response times. These tools, implemented as a collection of procedures, will help programmers to improve clarity and readability of simulation programs. The way they can be used has been inspired on the instrumentation capabilities of the CSIM simulation language. In particular, we have added four main structures to collect system data: `tables`, `qtables`, `boxes` and `meters`.

In this paper, the authors present a brief description of the implemented instrumentation tools and some aspects about their internal design. We also include a simple simulation example showing how to use them and how results are reported.

1 Introduction

Simulation is the imitation of the operation of a real-world process or system over time [1]. In particular, simulation is a useful technique for computer systems performance analysis. The choice of the programming language has a significant impact on the timely development of the model. General-purpose languages are portable and provide a good control over the efficiency and run time of the simulation [5]. Moreover, a model developed in a general purpose-language can be easily converted for execution on different computer systems.

SMPL (*Simple Portable Simulation Language*) [6] is a discrete-event simulation language. It is a functional extension of the C general purpose programming language, although implementations in other programming languages also exist, as Fortran, Pascal or PL/I. This extension takes the form of a set of library functions with, together with the programming language itself, compose an event-oriented simulation language. Simulation operations are performed by means of calls on the functions of the simulation subsystem. This approach provides a simulation capability suitable for small to medium scale models, whereas process-oriented simulation languages, like CSIM [2], are preferable for large scale models.

In a SMPL simulation model, there are three kinds of entities: facilities, customers and events [7]. A fa-

*This work was supported by the Spanish CICYT under grant TIC2000-0472-C03-03

cility is used to model a resource, such as the disk in a computer system or the network in a distributed system. In static form, a system comprises a collection of interconnected facilities. The interconnection of facilities is not explicit, but rather is determined by the routing of customers between them. Customers represent the active entities in the system. The dynamic behavior of the system is modeled by the movement of customers through the facilities defined in the model. Events represent a change of state of any system entity, such as the arrival of a customer into a facility. A SMPL simulation program comprises an initialization routine, a control routine, and some number of event routines.

Despite of SMPL was desinged many years ago, nowadays it continues being used at the universities of many countries in order to teach basic simulation matters due to its simplicity and easy use. For example, SMPL has been included as a part of the queue module of the SimPack toolkit [4].

Until now, we have found no improvement to that language in the literature since its original design. In this work we have added several instrumentation tools to SMPL for collecting and reporting distributions of simulation variables such as queueing times, queueing lengths, or response times. These tools, implemented as a collection of procedures, help programmers to improve clarity and readability of simulations programs. The added instrumentation tools have been inspired on these used in CSIM [2], a language for developing discrete-event, process-oriented simulation models. This fact allows experienced programmers in CSIM easily apply the same instrumentation procedures into SMPL programs. In particular, we have added four main structures to collect system data: **tables**, **qtables**, **boxes** and **meters**. More precisely, the extension of SMPL for the addition of tables has been originally discussed in [7]. Both the design and the data structure description for tables presented here have served as a starting point for own implementation.

The rest of the paper is organized as follows. Section 2 describes the functionality of each instrumentation structure added to the SMPL language. Section 3 shows some details about their implementa-

tion, and Section 4 describes and verifies the use of these tools by means of a simple simulation model. Finally, Section 5 summarizes the conclusions and future work.

2 Instrumentation Overview

In this section we examine the main features of the four implemented tools into the SMPL language. Most of the work in implementing these structures is in coding their report function; the code is simple, but the forming process becomes wearisome. Data values or events recorded in these structures are used to compute the statistics, and in all cases no data is actually stored. Previous to its use, each instrumentation tool added to the model must be initialized. All the data structures associated to these tools can be cleared by means of the `reset()` function.

Tables are used to collect statistics on a sequence of discrete values such as queueing times or response times. A value `v` is included in a table `t` using the procedure `record(v,t)`.

Qtables are appropriate to know the distribution of a population in a subsystem. A customer will execute `note_entry(qt)` and `note_exit(qt)` in order to note its entry and exit, respectively.

A **box** conceptually encloses a part of the model, reporting information about both the residence time and the population in that part of the model. This information is saved in both a **table** and a **qtable** associated to each **box**. A customer must execute `enter_box(b)` and `exit_box(b,ti)` to enter and exit a box, respectively, and it must specify the instant of time `ti` when entered the box.

A **meter** is used to collect statistics about the flow of customers at some point in a model. A customer notes its passage by a **meter** using the `note_passage(m)` procedure.

Statistics of the previous structures includes, among others, the mean, the number of values recorded, the minimum and maximum values, the sum and the range of values, etc. Also, confidence intervals for the mean can be automatically com-

puted by using the following formulas:

$$m \pm z_{1-\frac{\alpha}{2}} \frac{s}{\sqrt{n}}, \quad n > 30$$

$$m \pm t_{1-\frac{\alpha}{2}; n-1} \frac{s}{\sqrt{n}}, \quad n < 30$$

where n is the number of samples recorded, z and t are the normal and Student distribution functions, respectively, and $1 - \alpha/2$ is the confidence level or confidence coefficient (considered values in the implementation are 0.90, 0.95 and 0.98).

Finally, histograms can also be specified for the four instrumentation tools in order to get more detailed information about the collected values.

A detailed description of how to use these structures and how to gather statistics can be found in the CSIM User's Guide [2].

3 Implementation

The implementation of SMPL uses the C language. SMPL basic files are `smpl.c` (main simulation subsystem), `rand.c` (random variate generation functions), `smpl.h` (external name declarations) and `stat.c` (normal and t distribution quantile functions). In order to include the extra capabilities presented in this work, we have added code to the `smpl.c` and `smpl.h` files.

We have maintained the compactness of its original design, allowing both the use of SMPL in computers with limited memory sizes and as a part of a larger application-oriented simulation system. Compactness has been achieved using a single basic data element for all data structures. These elements comprises five fields, L1 and L2 for real values, and L3, L4 and L5 for integer values. SMPL maintains a pool of these elements, which are allocated from this pool in order to build facility descriptors, queue entries, and event list entries. This pool of elements has been used in order to construct `table`, `qtable`, `meter` and `box` descriptors. The basic structure of the element pool for the instrumentation tools is shown in Figure 3.

In the structure for tables we need to store the maximum and minimum of introduced values, as well as their sum and square sum, which will be used to obtain other statistics, such as the variance or the coefficient of variation. Additionally, there is a set of fields that are useful for the representation of the histogram of each table. These are "from" (it indicates the lower limit of the rank of the histogram), cell interval (it establishes the width of each interval), overflow and underflow (they count the values out of the rank), cell set index (it contains a reference to the first element of a structure, implemented as an array of integer type, that contains the number of observations that lie in each interval), and the number of intervals. Finally, there is a field to store the total amount of introduced values and a special field called "options", which indicates if the table belongs to a more complex structure like a box.

In the structure for `qtables` we need to consider two fields for keeping clock values: first, the instant in which a `qtable` starts to collect data, and also the instant when took place the last update. Moreover, it is necessary to consider the state of the `qtable` (length of the queue), the maximum and minimum values of this state, as well as the initial value of the queue length. As in the case of tables, `qtables` need to store the sum and square sum of values, the `qtable` chain link and also the index of name. Moreover, the number of entries and exits in the queue, as well as a group of fields that keep information related to the histogram, are considered. However, in the case of `qtables`, the cell set index links to an array of real type (each interval of the histogram represents a value of time).

The structure for meters have five fields: the clock value at the beginning of measuring, the clock value at the moment of the last update, the meter chain link and the index of name. The last field contains a value that identifies the table associated to the meter, because meters use functions offered by tables.

Finally, the structure for boxes is similar to that used by meters. However, in this case we need an additional field, because boxes use the functions provided by both tables and `qtables`. Therefore, this field keeps the identifier of the `qtable` associated to

Descriptor of tables:

maximum	minimum	overflow count	underflow count	unused
$\sum v$	$\sum v^2$	index of name	table chain link	cell set index
“from”	cell interval	options	no. values entered	no. of cells

Descriptor of qtables:

unused	options	cell intervall	no of cells	cell set index
overflow count	underflow count	initial value	“from”	exits
$\sum v$	$\sum v^2$	index of name	qtable chain link	entries
initial clock	last clock	state	minimum	maximum

Descriptor of meters:

initial clock	last clock	meter chain link	index of name	table
---------------	------------	------------------	---------------	-------

Descriptor of boxes:

unused	unused	unused	unused	qtable
initial clock	last clock	box chain link	index of name	table

Figure 1: Added elements to the SMPL basic pool

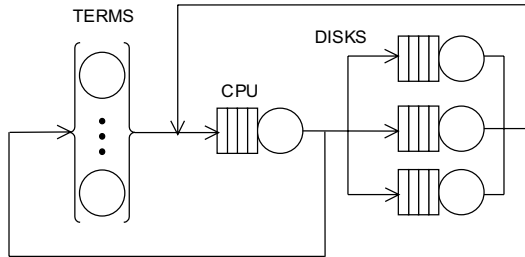


Figure 2: Queueing model

the box.

4 Simulation Example

In this section we present some results obtained using the SMPL simulation language with the added instrumentation tools. The model we have used is based on the central server, and it is depicted in Figure 2. This model has analytical solution that can

be found by using the Mean Value Analysis (MVA) algorithm [5]. This fact will help us to compare theoretical results with these one obtained by simulation. The parameters used for this model are shown in Table 1.

Parameter	Value
Processing time	0.05 s
Disk access time	0.035 s
Disk visit ratio	3
Thinking time	5 s
Number of disks	3

Table 1: Configuration parameters

We have added four monitoring points in this model, indicated in the Figure 3, in order to measure the mean values of the following system variables:

1. System response time. This variable measures

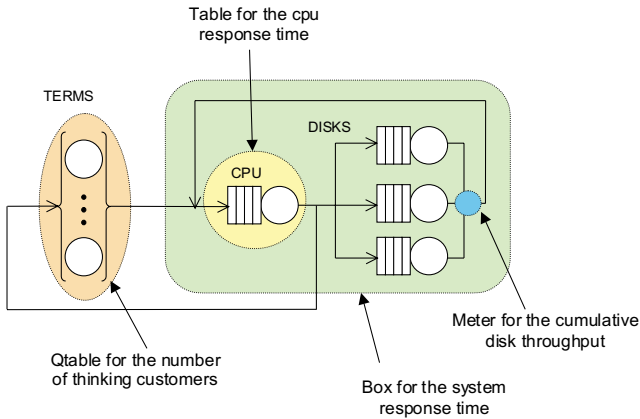


Figure 3: Queueing model indicating instrumentation points

the time a customer spends in the system, being computed by means of a **box**.

2. Throughput of disks. It measures the cumulative throughput of the three disks in the system, and it is measured by means of a **meter**.
3. Processor response time. This variable quantifies the time a customer spends in each visit to the CPU. It is measured by means of a **table**.
4. Number of thinking customers. It represents the number of customers in the thinking state (elapsed time between two consecutive requests to the system). We have measured it by using a **qtable**.

The following lines show the simulation program in SMPL language implementing the closed central server model. Comments has been added in order to provide enough readability to those who are not experienced with an event-oriented simulation language. A complete report of the results obtaining when running this program can be found in the Appendix.

```
#include "smpl.h"
#include <stdio.h>
#include <math.h>

#define MAX_JOBS 100
```

```
#define MAX_DISKS 5

double h, gm;
int start_period, batch_size;
int event, customer;
int i, endsim, nbatches;

double tcpu = 0.05;
double tdisk = 0.035;
double Zt = 5.0;
int JOBS = 15;
int DISKS = 3;

void sim(); // *** simulation function
void test_end(); // *** simulation end

struct info_token
{
    int disk;
    double arrival;
    double incpu;
} token[MAX_JOBS];

/* Facilities in the model. */
int TERMS, CPU, DISK[MAX_DISKS];

/* Instrumentation tools in the model. */
int b; // *** box
int m; // *** meter
int t; // *** table
int qt; // *** qtable
int bmt, gmt; // *** tables for batch means

/* Main program. */
void main()
{
    sim();
}

/* Simulation program. */
void sim()
{
    /* Initialization of model. */
    smpl(0, "Closed Model");

    /* Initialization of facilities. */
    TERMS = facility("TERMS", JOBS);
    CPU = facility("CPU", 1);
    for (i=0; i<=DISKS-1; i++)
        DISK[i]=facility("DISK", 1);

    /* Initialization of instrumentation tools. */
```

```

b = box("System response time");
box_time_confidence(b);
box_time_histogram(b,5,0.0,10.0);

m = meter("Cumulative throughput of disks");
meter_confidence(m);

t = table("Processor response time");
table_confidence(t);
table_histogram(t,3,0.0,0.75);

qt = qtable("Thinking customers");
qtable_histogram(qt,4,0,10);

bmt = table("Batch mean table");
gmt = table("Grand mean table");

/* Initial job scheduling. */
for (customer=1;customer<=JOBS;customer++)
    schedule(1,0.0,customer);

/* Initialization of batch means variables. */
start_period = 5000;
batch_size = 15000;
endsim=0;

while (endsim==0)
{
    cause(&event,&customer);

    switch(event)
    {
        case 1: // *** Begin of think time
            note_entry(qt);
            request(TERMS,customer,0);
            schedule(2,expntl(Zt),customer);
            break;

        case 2: // *** End of think time
            release(TERMS,customer);
            note_exit(qt);
            token[customer].arrival = enter_box(b);
            token[customer].incpu = time();
            schedule(3,0.0,customer);
            break;

        case 3: // *** Request cpu
            if (request(CPU,customer,0)==0)
                schedule(4,expntl(tcpcu),customer);
            break;

        case 4: // *** Release cpu
            release(CPU,customer);
            record(time()-token[customer].incpu,t);
            if (ranf()<=0.1)
            {
                exit_box(b,token[customer].arrival);
                test_end();
                schedule(1,0.0,customer);
            }
            else
            {
                token[customer].disk=random(0,DISKS-1);
                schedule(5,0.0,customer);
            }
            break;

        case 5: // *** Request disk
            i = token[customer].disk;
            if (request(DISK[i],customer,0)==0)
                schedule(6,expntl(tdisk),customer);
            break;

        case 6: // *** Release disk
            i = token[customer].disk;
            release(DISK[i],customer);
            note_passage(m);
            token[customer].incpu = time();
            schedule(3,0.0,customer);
            break;
    }
}
full_report();
printf("\nResponse time: %f +/- %f",gm,h);
}

void test_end() // *** batch means
{
    if (table_cnt(box_time_table(b))>start_period)
    {
        record(time()-token[customer].arrival,bmt);
        if (table_cnt(bmt)==batch_size)
        {
            record(table_mean(bmt),gmt);
            nbatches = table_cnt(gmt);
            gm = table_mean(gmt);
            h = table_conf_halfwidth(0.95,gmt);
            if ((nbatches>=10)&&((h/gm)<0.1)) endsim=1;
            reset_table(bmt);
        }
    }
}
}

case 4: // *** Release cpu

```

Instrumentation tools in the model are variables

b, **m**, **t** and **qt**. Confidence intervals and histograms have been requested for variables **b** and **t**.

The number of thinking customers can be computed by noting the entry into the **TERMS** facility (case 1) and noting the exit when releasing it (case 2). The mean value of thinking customers calculated by the **qtable** is 9.4656, agreeing with that indicated by the analytical solution.

The system response time lasts since a job leaves the **TERMS** facility (case 2) until it releases the processor to begin a new think time (case 4). These two cases are used to note the entry and the exit, respectively, of a job into the **box**.

Throughput of disks has been measured by noting the passage of each job when releasing any of the three disks in the model (case 6).

Processor response time is calculated using a **table** as the difference between the time a job releases it (case 4) and the time that job entered it (case 2). The computed mean response time is 0.2539 ± 0.003 (95% confidence level), that agrees the analytical solution.

The simulation method used is batch means [3], and it has been implemented in the **test_end()** function similar to the batch means module designed in [7]. However, in our case, the method has been implemented using two **tables**. The output simulation value used is the mean system response time. It has been defined a start period of 5000 observations and each batch consists of 15000 samples. After the initial period, values of the variable are considered until a full batch is accumulated in the **bmt** table. Then, the mean of the batch is considered to compute the grand mean by using the **gmt** table. The simulation stops when ten or more batches have been collected and the relative error between the grand sample mean and the half width of the confidence interval is less than 0.1. The confidence interval is computed with a confidence level of 95 percent. The simulation of this model gives a system response time of 2.929 ± 0.051 , that includes the theoretical value of 2.928.

5 Summary and future work

In this paper we have described a set of instrumentation tools that have been added to the well-known SMPL language. These tools can help model designers to implement simulation programs easier to read, debug and understand. The added source to SMPL is open and it can be obtained by sending an e-mail to one of the authors of this work.

As a future work, we are now working on the inclusion into SMPL of another set of extensions such as: storages (to model resources), facilities with infinite servers (to model delays), customer classes (for multiclass systems), and other scheduling policies at queues (as Round Robin).

References

- [1] J. Banks (ed.), *Handbook of simulation. Principles, Methodology, Advances, Applications and Practice*, McGraw-Hill, 1991.
- [2] *User's guide: CSIM18 Simulation Engine (C version)*, Mesquite Software, Inc.
- [3] A.M. Law and W.D. Kelton, *Simulation modeling and analysis*, McGraw-Hill, 1991.
- [4] P.A. Fishwick, *Simulation model design and execution. Building digital worlds*, Prentice Hall, 1995.
- [5] R. Jain, *The art of computer system performance analysis*, John Wiley & Sons, Inc., 1991.
- [6] M.H. MacDougall, *SMPL – a simple portable simulation language*, Amdahl Corp., Technical Report, April 1, 1980.
- [7] M.H. MacDougall, *Simulating computer systems. Techniques and tools*, The MIT Press, 1987.
- [8] H. Schwetman and J. Brumfield. Data analysis and automatic run-length control in CSIM18, *Proceedings of the 1997 Winter Simulation Conference*, pp. 687–692. 1997.

Appendix: full report of simulation

This appendix contains the output of the simulation program presented in Section 4. This output is produced by means of the `full_report()` function.

smpl SIMULATION FULL REPORT						
Closed Model						
		Ending simulation time:		82057.641		
		Elapsed simulation time:		82057.641		
facility name	service time	util.	through- put	queue length	response time	compl count
TERMS	5.0105	9.4647	1.8890	9.4647	5.0105	155004
CPU	0.0501	0.9457	18.8940	4.7981	0.2539	1550399
DISK[0]	0.0350	0.1991	5.6836	0.2462	0.0433	466386
DISK[1]	0.0350	0.1980	5.6573	0.2446	0.0432	464228
DISK[2]	0.0351	0.1986	5.6641	0.2454	0.0433	464784

TABLE : Processor response t

minimum	0.0000	mean	0.2539
maximum	1.5154	variance	0.0293
range	1.5154	standard deviation	0.1712
observations	1550399	coeficient of var	0.6741

confidence intervals for the mean after 1550399 observations

level	confidence interval			
90 %	0.2539 +/-	0.0002 = [0.2537,	0.2542]
95 %	0.2539 +/-	0.0003 = [0.2537,	0.2542]
98 %	0.2539 +/-	0.0003 = [0.2536,	0.2543]

HISTOGRAM Processor response t

lower limit	frequency	proportion	cum. prop.	
0.0000	858129	0.55349	0.55349	*****
0.2500	550598	0.35513	0.90862	*****
0.5000	125825	0.08116	0.98978	**
> 0.7500	15847	0.01022	1.00000	

TABLE : Batch mean table

>No data recorded since creation or reset

TABLE : Grand mean table

minimum	2.7836	mean	2.9290
maximum	3.0286	variance	0.0050
range	0.2449	standard deviation	0.0707
observations	10	coefficient of var	0.0241

QTABLE : Thinking customers

initial	0	minimum	0	mean	9.4656
final	10	maximum	15	variance	7.1721
entries	155014	range	15	standard deviation	2.6781
exits	155004			coefficient of var	0.2829

HISTOGRAM Thinking customers

number	total time	proportion	cum. prop.	
0	300.6090	0.00366	0.00366	
3	5872.4408	0.07157	0.07523	**
6	23425.0911	0.28547	0.36070	*****
> 9	52458.9600	0.63930	1.00000	*****

METER : Cumulative throughput of disks

count	1395398	rate	17.0051
-------	---------	------	---------

interpassage time statistics

minimum	0.0000	mean	0.0588
maximum	3.2740	variance	0.0042
range	3.2740	standard deviation	0.0652
observations	1395398	coefficient of var	1.1082

confidence intervals for the mean after 1395398 observations

level	confidence interval			
90 %	0.0588 +/-	0.0001 = [0.0587,	0.0589]
95 %	0.0588 +/-	0.0001 = [0.0587,	0.0589]
98 %	0.0588 +/-	0.0001 = [0.0587,	0.0589]

BOX : System response time

statistics on elapsed time

minimum	0.0000	mean	2.9299
maximum	47.2148	variance	9.8671
range	47.2148	standard deviation	3.1412
observations	155000	coefficient of var	1.0721

confidence intervals for the mean after 155000 observations

level	confidence interval				
90 %	2.9299 +/-	0.0131 = [2.9167,	2.9430]	
95 %	2.9299 +/-	0.0156 = [2.9142,	2.9455]	
98 %	2.9299 +/-	0.0186 = [2.9113,	2.9484]	

HISTOGRAM

lower limit	frequency	proportion	cum. prop.	
0.0000	80445	0.51900	0.51900	*****
2.0000	36211	0.23362	0.75262	*****
4.0000	17780	0.11471	0.86733	****
6.0000	9358	0.06037	0.92770	**
8.0000	4943	0.03189	0.95959	*
> 10.0000	6263	0.04041	1.00000	*

statistics on population

initial	0	minimum	0	mean	5.5344
final	4	maximum	15	variance	7.1721
entries	155004	range	15	standard deviation	2.6781
exits	155000			coefficient of var	0.4839

System response time: 2.929 +/- 0.051

HARDWARE IN THE LOOP SIMULATION USING REAL-TIME CD++

Lidan Li
Gabriel A. Wainer
Trevor Pearce

Department of Systems and Computer Engineering
Carleton University
1125 Colonel By Drive
Ottawa, ON, K1S5B6, Canada
Email: {lidan, pearce, gwainer}@sce.carleton.ca

KEYWORDS

DEVS formalism, real-time simulation, CD++.

ABSTRACT

Modeling and simulation tools have been used for helping in the early stages of hardware/software systems design. The DEVS formalism is a technique that enables hierarchical description of discrete event models that can be used for this task. The CD++ tool enables the description of discrete event models based on the DEVS formalism, and we have used it to provide hardware-in-the-loop simulation using the CODEC of a DSP board. First, a DEVS model was built using the real-time version of CD++ to simulate the behavior of the CODEC together with a test program using the CODEC. Next, the actual CODEC was deployed as a hardware prototype to replace the CD++ model, integrating the prototype into the original DEVS component. The real-time data communication between the CD++ model and the DSP board was explored in detail. As a result, we are now able to study models in a simulated environment, and to execute them in a hardware surrogate. The hierarchical nature of DEVS permitted to do this without modifying the original models, providing the base for enhanced system development in embedded platforms.

INTRODUCTION

In the early stage of the design of embedded systems, software and hardware are designed independently. The software development team is waiting for the hardware prototypes to be available; however, the hardware development team is waiting for the software environment for hardware prototype verification and testing. It is difficult to decide the trade-offs between the hardware and software solutions in terms of system performance requirements (time, power consumption) and probably delays the product design cycle. In addition, there are few interactions between hardware and software, which restrict the exploration of solutions where some functionality could migrate between both components (Berge 1997). With the emerging of the specific components (I/O, DSP, ASIC, FPGA), a mixed-system design is more efficient to realize specific applications, such as signal processing or telecommunications. The user can test the functionality of the hardware in a very early stage. This is economically

efficient, and shortens the product development cycle and time-to-market period.

The DEVS (Discrete EVents Systems specifications) formalism for modeling and simulation (Zeigler, Praehofer and Kim 2000) provides a framework for the construction of hierarchical models in a modular fashion, allowing model reuse, reducing development and testing time. The hierarchical and discrete event nature of DEVS makes it a good choice to achieve an efficient product development test. DEVS are timed models, which also enables us to define timing properties for the models under development. Each DEVS model can be built as a behavioral (atomic) or a structural (coupled) model. A DEVS atomic model is described as:

$$M = \langle X, S, Y, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, D \rangle$$

X: the input events set
S: the state set
Y: the output events set
 δ_{int} : internal transition function
 δ_{ext} : external transition function
 λ : output function
D: the elapsed time

A DEVS coupled model is formed by configuring several atomic models or coupled models:

$$CM = \langle X, Y, D, \{M_i\}, \{I_i\}, \{Z_{ij}\}, \text{select} \rangle$$

X: the set of input events
Y: the set of output events
D: an index of components, each $i \in D$
 M_i : a basic DEVS model, where $M_i = \langle I_i, X_i, S_i, Y_i, \delta_{\text{int}_i}, \delta_{\text{ext}_i}, \tau_{a_i} \rangle$
 I_i : the set of influencees of model I_i , each $j \in I_i$
 Z_{ij} : the i to j translation function
Select: the function prescribes which atomic model should be activated first under simultaneous events.

The CD++ environment (Wainer 2002) is a tool built to implement the DEVS and Cell-DEVS theory. The toolkit has been built as a set of independent software pieces, each of them independent of the operating environment chosen. The defined models are built as a class hierarchy, and each

of them is related with a simulation entity that is activated whenever the model needs to be executed. New models can be incorporated into this class hierarchy by writing DEVS models in C++, overloading the basic methods representing DEVS specifications: external transitions, internal transitions and output functions. CD++ employs a virtual time simulation approach (Rodríguez and Wainer 1999), which allows skipping periods of inactivity. The abstract simulation technique enables defining and using different simulation engines without affecting existing models. The recent real-time extension (Glinsky and Wainer 2002) enables simulation advancing based on the wall-clock, making the simulation process to be quiescent between events. The model being executed must react to external event in a timely fashion. This means when an external event arrives, the model should react within a predefined deadline, and return a result before that time. The real-time extension of the toolkit allows associating deadline with external events.

We show how to gradually incorporate hardware prototypes into a simulated environment using the real-time version of CD++ toolkit (Glinsky and Wainer 2002). The hardware prototype employed is a CODEC (a device performing A/D and D/A operations) embedded in a Digital Signal Processor (DSP) board (Analog Devices 2000). The block diagram in figure 1 below is the 2189M EZ-KITLITE DSP Board (Analog Devices 2000) and some major components used.

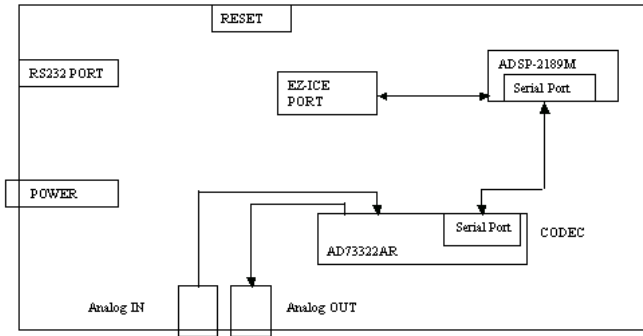


Figure 1: Scheme of the Analog Devices 2189M EZ-KITLITE Evaluation Board

THE APPLICATION MODEL

The prototype was designed in two stages. In the first stage, we built a set of DEVS atomic models using the real-time version of CD++. Figure 2 shows the *test A* experimental frame built in CD++, which was used to test the behavior of the CODEC model.

The *test A* experimental frame includes five atomic models. They are *clock*, *control*, *CODEC*, *analog signal generator* and *display*. A brief description of the these atomic models is given below:

- **Clock**: generates control signals with a predefined period.

- **Control**: distinguishes the incoming signal. If a command signal is received, it will invoke the CODEC atomic model.
- **CODEC**: simulates the behavior of the CODEC.
- **Analog signal generator**: generates an analog signal.
- **Display**: updates the results from the CODEC model and displays them.

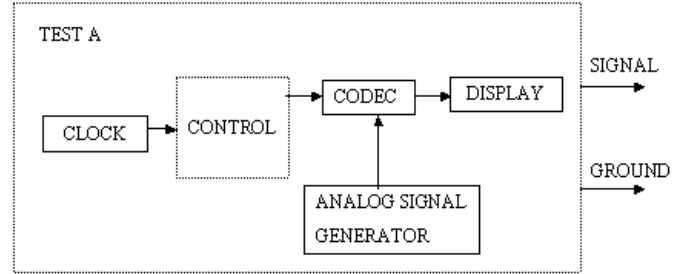


Figure 2: Test A, Experimental Frame Conceptual Model

The detailed definition of these models can be found in (Li, Pearce and Wainer 2002), but we will focus in the description of the *CODEC* model. Later, we will show its implementation using the available hardware. The *CODEC* model is responsible for translating an analog signal into an output digital signal. Figure 3 shows the CODEC DEVS atomic model used in our example.

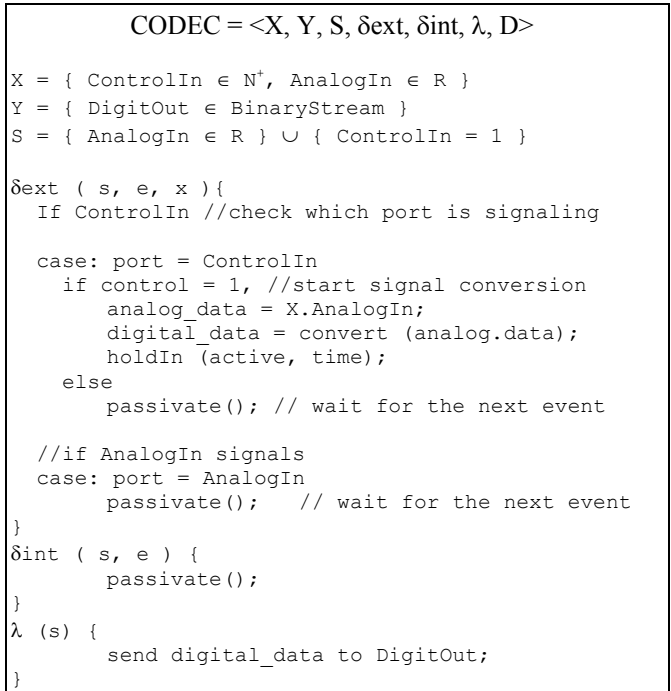


Figure 3: CODEC Model Conceptual Definition

When an event arrives at *ControlIn* and its value equals to 1, the CODEC atomic model starts analog to digital signal conversion using the analog received in the *AnalogIn* port. When the signal conversion completes, the CODEC sends all results using the *DigitOut* port. The analog signal is a stream of floats and the digital signal is a stream of binary

strings. Please see the table 1 for a sample analog->digital conversion. The Analog generator is an atomic model and all it does is to randomly generate float representation of the analog signal.

Table 1: Analog to Digital Conversion

Analog Input (voltage)	Digital Output
0	0000
0.2	0001
0.4	0010
0.6	0011
0.8	0100

The *test A* coupled model is built to test the behavior of the software version of the CODEC. Figure 4 is the simulation output of the *test A* experimental frame. According to the figure, the first digital signal is obtained at time 00:00:40:708 through the *signal* output port. Its associated value is 0001 (here we use binary strings representing the digital signal value). At the time 00:00:41:608, another digital signal is arrived at *ground* output port and its value is 0000. All digital signals are processed successfully through the software CODEC model.

Wall Clock Time	Result	Output Port	Value
00:00:40:708	succeeded	signal	0001
00:00:41:008	succeeded	signal	0110
00:00:41:308	succeeded	signal	0011
00:00:41:608	succeeded	ground	0000
00:00:41:900	succeeded	signal	1010
...			

Figure 4: Output of Test A Experimental Frame

INTRODUCING HARDWARE-IN-THE-LOOP

The application model we built in the previous section was reused when we replaced the model for the CODEC by the actual hardware. The new experimental frame (figure 5) was built to support interaction with the real CODEC on the board, and existing atomic models were reused. A *clock* model generates periodic signals to awake the *control* model. Then the *control* model will invoke the *TCL* model, which is in charge of initializing the CODEC and start the conversion. When the CODEC finishes the conversion, the *dataTransfer* model will acquire these data and send them to the *control* model. Finally, the *DataTransfer* model will feed the data to the *display* model. A brief description of the new atomic models is given following:

- **TCL**: invokes and opens the VisualDSP debugger system. Once the debugger system is opened, the different TCL files needed for A/D and D/A access can be invoked to obtain samples and regenerate the analog signal.
- **DataTransfer**: reads the digital samples and sends them back to the control model. In addition, the data will be written back to the board for display if the DAC is working.

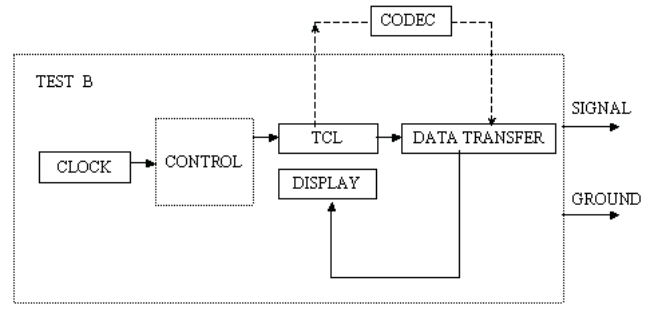


Figure 5: Test B Experimental Frame Conceptual Model with Hardware-in-the-loop

Figure 6 shows the CD++ coupled model definition of the *Test B* experimental frame. The first line in the figure defines the *Top* model, which includes four atomic models and one coupled model. The *Out* port represents two output ports: signal, ground, and the links describe the internal and external coupled schema. The similar configurations are specified for each of the atomic models in the Test B experimental frame conceptual model.

The **[top]** model always defines the coupled model at the top level. As showed in the formal specifications presented earlier, four properties must be configured:

- **Components**: describes the models integrating a coupled model. The syntax is `modelName@className`, allowing more than one instance of the same model using different names. The class name reference to either atomic or coupled models (which should be defined in the same configuration file).
- **Out**: it defines the names of output ports.
- **In**: it defines the names of input ports.
- **Link**: it describes the internal and external coupling scheme. The syntax is: `source_port[@model]` `destination_port[@model]`. The name of the model is optional and, if it is not indicated, the coupled model being defined is used.

```
[top]
components : clock@Clock control@TclCycle
display@DisplayCycle DataTransfer@FileTransferCy
Out : ground signal
Link : out@clock in@control
Link : out@control in@tcl
Link : out@tcl start@transfer
Link : out@transfer in@display
Link : ground@display ground
Link : signal@display signal

[control]
components : queue@Queue central@ControlCycle
In : in
Out : out_data_out
Link : in in@queue
Link : out@queue in@central
Link : data_out@central done@queue
...
```

Figure 6: CD++ Coupled Model Specification for Source Code for *Test B*

IMPLEMENTATION

The system was designed using two components: one running in a PC/Workstation, and another on the DSP board. See figure 7.

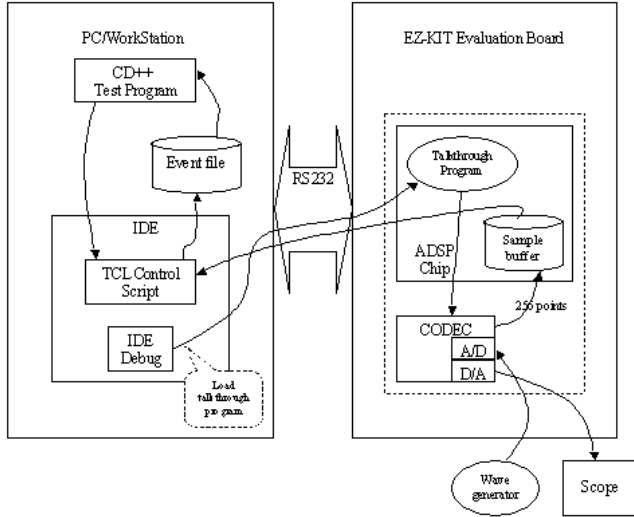


Figure 7: A High-level Architecture of the Simulation

The CD++ test experimental frame communicates with the board through the IDE software, and a built-in TCL script controls the debugger operation. The IDE uses serial communication to access the ADSP chip memory on the DSP board. In the board, a *talkthrough* program is loaded by the IDE, which invokes the CODEC, stores the digital samples in a circular buffer, and sends the data back to the CODEC for output. When the samples are available, the TCL control script gets them and makes them available for CD++ as new external events. The interaction between the components is depicted in Figure 7. There are three cooperating subsystems in the simulation, the DSP board, the IDE and CD++. The DSP board subsystem acquires an analog signal from a 15MHZ function/waveform generator, and digitizes it through an A/D of the CODEC. These digital samples are saved in a predefined location in the DSP chip memory. In addition, the CODEC D/A writes these digital samples back to reproduce the analog signal, and displays them in a digital oscilloscope.

An important issue raised was how to achieve the system's deadline. The real-time CD++ simulator uses the wall-clock to determine when to execute the next event. In this case, when the TCL model invokes the IDE application, we need time to reset the board. The solution is to let the model to wait for the hardware to start. During this period, the model will remain in the active state, and after that, it will generate an internal transition, recording the physical time it takes to this action. Another issue is to ensure that the model is ready to start a new read cycle, and that will depend on the speed for reading. The clock model will generate the read command according to the following diagram (Figure 8):

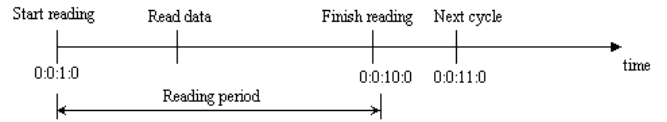


Figure 8: Clock Timing Diagram

Figure 8 shows that at time 0:0:1:0, the model started to read data from the board memory. After 9 time units, the model finished reading the data. The clock model will record this time, and then it will issue the next command at time 0:0:11:0 to start another cycle.

Displaying the analog signal on the real-time digital oscilloscope is another difficult issue. Due to the limitation of the current IDE debugger, only 256 sample points can be written back at a time. These digital points went through the D/A and displayed on the oscilloscope. The following digital pictures show a 1 KHz sine waveform on the scope.

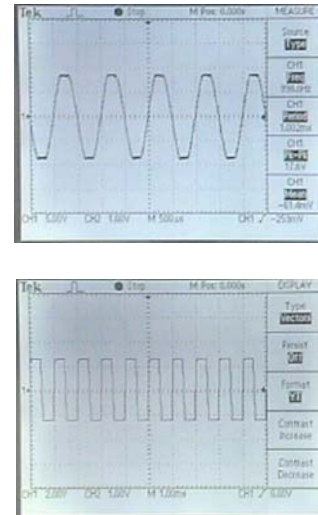


Figure 9: Digital Pictures of 1 KHz Sin Waveform

The top is the input signal in sine waveform; the bottom image is the output from the simulation. Since the output signal is instantaneous, the display of the output signal is caught by freezing the screen. After adjusting the scale of the signal, we can read that the frequency is 1 KHz, which means the output signal is correct (in the picture they seem to be different, because the digital oscilloscope does not have enough signals to acquire in order to auto-set the display).

TEST EXAMPLES

Different tests were conducted to evaluate the software and hardware components, as well as the program as a whole (refer to Test B conceptual frame) to examine the correctness and performance of the simulation.

1) Examining the A/D functionality of the CODEC. At initialization, the clock atomic model generated a control signal to start the *control* atomic model at time 00:00:00:040. The control model then invokes the TCL

model to start the IDE application and to load the TCL script. After pressing the *reset* button on the DSP board, the *talkthrough* program is loaded into the DSP board. At the same time, the *dataTransfer* atomic model is hold in the sleep state and is waiting for the data to be generated by the CODEC on the DSP board. Upon receipt of the converted digital data, the *dataTransfer* model will take these data and send them back to the control model. Finally, through a display atomic model, these samples will be updated and stored in a bin. After one simulation cycle, the clock will generate the next command. The simulation can be terminated by a preset simulation time or manually.

The following execution results show that all the signal conversions are successful and the values are obtained. These results match with those of the *test A* experimental frame and the differences in the value column represent a difference in the format and the analog waveform used.

Wall clock time	results	Output Port	value
00:00:40:708	succeeded	signal	0.50391
00:00:41:008	succeeded	signal	-1.00000
00:00:41:308	succeeded	signal	-0.97546
00:00:41:608	succeeded	ground	0.03967
00:00:41:900	succeeded	signal	-1.00000
...			
00:06:51:210	succeeded	signal	0.99997
00:06:51:210	succeeded	ground	0.03894

Figure 10: Output of Test Model

2) D/A functionality of the CODEC was added into the simulation cycle. The purpose of this test is to verify the correctness of the digital samples obtained from the previous testing case.

CONCLUSION

We presented how to achieve hardware-in-the-loop simulation of discrete event models based on the DEVS formalism. The real-time data communication between the CD++ model and the DSP board was explored in detail. As a result, we are now able to study models in a simulated environment, and to execute them in a hardware surrogate. The hierarchical nature of DEVS permitted to do this without modifying the original models, providing the base for enhanced system development in embedded platforms.

This is the first version of the CD++ simulation platform that employs an actual piece of hardware in the simulation loop. The hardware has been successfully introduced into the CD++ simulation platform. Different test examples and results were described to verify the correctness of the approach. The future development could be done towards making large systems and exploring different ways of hardware/software communications to further enhance and formalizing real-time simulation using CD++.

REFERENCES

Analog Devices, 2000. "ADSP-2189M EZ-KIT Lite Evaluation System Manual".

Analog Devices, 2000. "AD73322 data sheet". Analog Devices Technical Staff.

Analog Devices, 2000. "ADSP-218x DSP Hardware Reference".

Analog Devices, 2000. "VisualDSP User's Guide for ADSP-21xx DSPs".

Berge, J.M. 1997. "Hardware/Software Co-Design and Co-Verification". Kluwer Academic Publishers.

Glinisky, E.; Wainer, G. 2002. "Performance Analysis of Real-Time DEVS models". In *Proceedings of 2002 Winter Simulation Conference*. San Diego, U.S.A.

Li, L.; Pearce, T.; Wainer, G. 2002. "An experience in hardware-software codesign using the DEVS formalism". Technical Report SCE-12-02, Department of Systems and Computer Engineering, Carleton University.

Rodríguez, D.; Wainer, G. 1999. "New Extensions to the CD++ tool". In *Proceedings of 31st SCS Summer Computer Simulation Conference*. Chicago, U.S.A.

Wainer, G. "CD++: a toolkit to define discrete-event models". 2002. In *Software, Practice and Experience*. Wiley. Vol. 32, No.3, pp. 1261-1306.

Zeigler, B.; Praehofer, H.; Kim, T. 2000. "Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems". Academic Press.

BIOGRAPHIES

Lidan Li is a M.A.Sc. student in the Department of Systems and Computer Engineering (SCE) at Carleton University. She received her B. Eng. degree (1998) from Shanghai University, Shanghai, China.

Gabriel Wainer received a M.Sc. (1993) and a Ph.D. (1998, summa cum laude) in Computer Sciences from the Universidad de Buenos Aires (Argentina) and Université d'Aix-Marseille III (France). He is Assistant Professor in the SCE Department, Carleton University. He was Assistant Professor at the Universidad de Buenos Aires, Argentina, a visiting research scholar at ACIMS, University of Arizona and LSIS/CNRS (Marseille, France). He is author of two books and numerous articles on Discrete-Event simulation and real-time systems. He is Associate Editor of the Transactions of SCS. Prof. Wainer was a member of the Board of Directors and he is the chair of Standards Committee of the SCS. He is Associate Director of the Ottawa Center of The McLeod Institute of Simulation Sciences, and the coordinator of an international group on DEVS standardization.

Trevor Pearce completed both a B.Eng. (with High Distinction, 1982) and a M.A.Sc. (1986) in Electrical Engineering at Carleton University, and a Ph.D. in 1994, at Queen's University with the Department of Computing and Information Science. He has been an Assistant Professor at Carleton University, in the SCE Department since 1995, and has served a term as the Associate Chair for Undergraduate Studies. He is researching embedded systems development methods based on modeling and simulation, real-time simulation, and distributed simulation based on the High Level Architecture (HLA). At Carleton, he is a member of the RADS Lab, and a founding member of the Embedded Systems Group. He is a member of OC-MISS, and the founding Chair of the Standards Activities Committee for SISO Canada.

DISTRIBUTED PROCESS SIMULATION USING OPC: A CASE STUDY

Raúl Alves and Julio E. Normey-Rico
Department of Automation and Systems
Federal University of Santa Catarina
88040-900 CTC-UFSC
Florianópolis-SC Brazil
E-mail: {ralves,julio}@das.ufsc.br

Alejandro Merino
Center of Sugar Technology
University of Valladolid
Edificio Alfonso VIII. 47011
Valladolid, Spain
E-mail: alejandro@cta.uva.es

Luis Felipe Acebes and Cesar de Prada
Department of System Engineering and Automatic Control
University of Valladolid
Prado de la Magdalena s/n 47011, Valladolid, Spain
E-mail: {felipe,prada}@autom.uva.es

KEYWORDS

Training, Industrial processes, Distributed processors, Communications processors, Interactive simulations, Simulation interfaces.

ABSTRACT

This paper shows a methodology to solve a problem of execution speed of a continuous processing plant dynamic simulation that acts as data engine for a personnel-training system for the sugar industry. This system is based on a set of PCs over Windows NT linked by a communication network, uses a Supervisory Control and Data Acquisition system (SCADA) to emulate the control rooms normally used in sugar factories (additionally this SCADA can be configured as a trainer console) and substitutes the production process for the simulation previously mentioned. The main idea of the developed methodology is to distribute the simulation in several simulation engines that are synchronized and share data using the OPC communication protocol. This methodology, that can be applicable to any full scale simulator, is illustrated using the training simulator developed in the CTA (Sugar Technology Center) of the University of Valladolid (Spain).

INTRODUCTION

Simulation systems are very important as industrial training tools. The objectives of the simulation program is to reproduce in the most reliable way the global dynamic performance of the process and to allow the training of operators in abnormal and dangerous situations. In the case of sugar production the process can be divided on several stages; in each one some process units appear repeatedly. For this reasons, a library of complex mathematical first principle models of process units has been developed using an Object Oriented Modelling Language for hybrid systems named EcosimPro (EcosimPro, 2003). Using this library is possible to construct a dynamic model of a whole sugar factory and simulate its behaviour.

In order to train the operators in the same conditions as in the real plant the dynamic simulation must run at least in real time. However, the complexity of the mathematical models of the process units (the plant is composed by continuous, batch and semi batch process units) and the high number of them (one factory can has several hundreds of process units) do not allow to work in real time using traditional simulation procedures.

As, it is not possible to manipulate the ODEs integration algorithms and the event treatment laws of the simulation package, only two procedures can be used to increase the speed of the simulation. The first one is to modify the library of models and reduce their complexity. The second one is to distribute the whole simulation in several computers. As reducing the complexity of the models implies in a reduction of the reliability of the simulations the second alternative seems to be the most indicated. Thus, this paper presents the development of a distributed simulator for a sugar factory. The paper is organised as follows: in next section some alternatives are shown and the standard that they apply. Section 3 describes the distributed architecture of the simulator based on OPC. Section 4 presents the application of the methodology to a sugar plant simulator. Finally, in section 5 some conclusions are presented.

DISTRIBUTED SIMULATION

In industrial environments large scale control systems uses communications at different levels: at low or field level, at medium or control-supervision level and at high or management level. In the low level, industrial field buses are normally used to perform the communications between devices. In the others levels traditionally specific drivers are used to allow the access to the devices through the field bus. In order to avoid the use of a great number of specific drivers in the same network, a group of companies, in collaboration with Microsoft, define a standard for the data intercommunication called OPC (for OLE Process Control) (F.Iwainitz et al, 2002). This standard is based on a Client-

Server communication system and allows a uniform access to the data connected to a network.

In Microsoft Windows environments, the communications between distributed components are based on DCOM (Distributed Component Object Model) (Microsoft, 1996). DCOM allows the development of software components using any programming language and network protocol and also with free location. CORBA (Common Object Request Broker Architecture) (OMG, 1998) and Java RMI (Remote Method Invocation) (Sun, 2003) can also be used instead of DCOM as they have similar characteristics. The objective of all these middlewares is to provide a software bus on which components and clients can be connected independently origin and location.

OPC is based on DCOM. It consists in a standard set of interfaces, properties and methods that can be used in process control and manufacture applications. Its main objective is to uniform the access to the data in industrial applications, in such a way that several clients can access to the data managed by an OPC server through a Windows network (Figure 1).

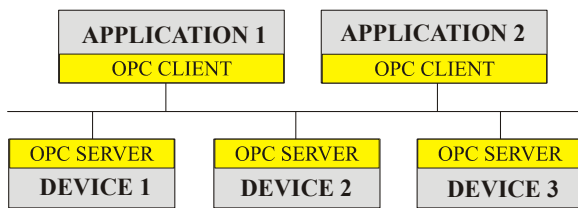


Figure 1. Client-Server communications with OPC.

For the distribution of simulations it is possible to use standards like HLA (High Level Architecture) (DoD DMSO, 2002) and CAPE-OPEN (Computer Aided Process Engineering Open Simulation Enviroment) (CAPE-OPEN, 1999). HLA uses their own communications and synchronization libraries, called Runtime Infrastructure (RTI). CAPE-OPEN specifies the interfaces of the components for CORBA and DCOM. In both cases, the common idea is the accomplishment of a series of components specifications so that they can be developed and easily reused by third people.

On the practical case studied in this paper, the use of OPC is based on the necessity to uniform the access to the simulations, as well as to facilitate the remote access to them. The system will allow the access to the simulations and OPC servers from supervision and control systems. Thus, defining an OPC server interface for the simulations, it is possible to connect the simulations with several control and supervision software (commercial or local developed software) and also to use remote access.

Also the OPC interface can be used in the distributed simulation to make the data interchange and synchronization avoiding the use of additional communication mechanism.

DISTRIBUTED ARCHITECTURE OF THE SIMULATOR BASED ON OPC

The main property of OPC servers is that they allow the access from any OPC application client. In this work the objective is to provide an OPC server interface to the simulations generated by Ecosimpro (Zamarreño et al., 2000), in such a way that any client can access them in the same way as other OPC server associated to an industry physical device.

The OPC interface can be added to the simulations because Ecosimpro could generate the code of the simulation in C++ language. The generation of the servers is made by the union of the simulation code and the OPC server, which is created using specific toolbox of the company Softing (Softing, 2002) that facilitates the development of OPC servers and clients.

The server has two principal parts: the thread that handles the advance of the simulation, and the OPC server, which includes the access to the simulation (Figure 2).

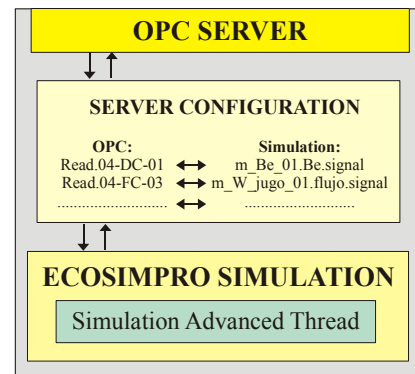


Figure 2. Parts of the server.

For the generation of the servers with the simulations a program that automates the process has been developed. In order to generate the server is enough filling up a series of fields relatives to: the simulation in Ecosimpro, where the server will be generated, as well as some parameters for its configuration. Once generated the server it is necessary to generate a configuration file indicating the relation between the OPC server variables and the simulation variables.

When simulating a complete process, it is necessary to communicate several simulations running in different computers of the network.

A program that makes the data interchange between all the OPC server simulations does this communication. Also, to change the data in the some sample period a synchronization mechanism must be used (Figure 3).

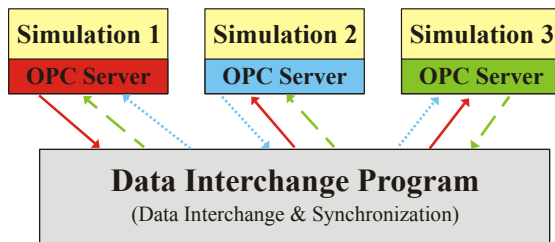


Figure 3. Data exchange between simulations.

In the proposed system an external application interchanges the data and makes the synchronization of the simulations. This application includes an OPC client configured with all the variables to interchange and the control variables of the

simulations. This application contains a thread that handles the interchange of data between the simulations. When it detects that all the simulations have finished their calculations makes the data interchange, after this procedure is finished, it orders to the simulations to make the next calculation.

The resulting architecture of the simulator consists of several PCs connected in a Windows NT network. All of them execute the simulations, and one of them is used to run the program that makes the union of the simulations (Figure 4).

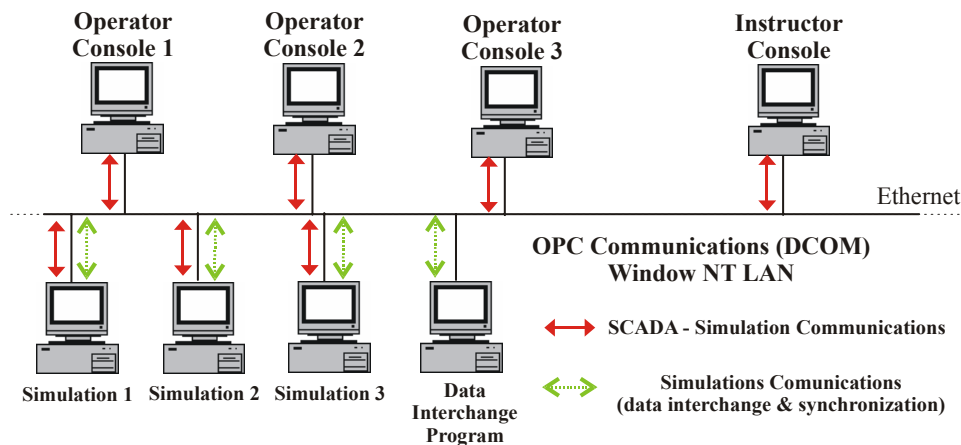


Figure 4. Simulator architecture.

CASE STUDY: APPLICATION TO A SUGAR PLANT SIMULATOR

This methodology it has been used in the CTA (Sugar Technology Centre) for the development of a sugar factory training simulator.

From a functional point of view the simulator has a real time simulation of the sugar production process and its control system, one or several operators consoles from which the operator acts over the simulation as he was acting over the real process, and an instructor console from which the supervisor manages the training session and projects schematics, photographs and videos related with the simulated process.

In sugar factories, sucrose crystals are obtained from sugar beets. Sugar beets are cut and washed. Once this has been done the extraction is carried out in the diffusion section. In this section we extract the sucrose contained in the sugar beets using hot water. We obtain a dissolution of sucrose and other substances called juice. Impurities present in this juice are removed in the depuration stage. After that, the juice is concentrated by means of the evaporation of part of the water contained in the juice. The concentrated syrup obtained is crystallized in the sugar house. Then, crystals

are dried and packed, and ready for selling. Exhausted beets, with a previous drying stage, are used as animal food. An scheme of the production process is shown in the figure 5.

Due to the calculus load that would originate a whole plant simulation in a single computer the plant model has been divided in six different stages interchanging data one each other. In sugar plants there are points, usually connecting different kind of operations, in witch only two or three pipes do the connection. For example between diffusion stage and depuration stage there is only one pipe connecting them, a pipe with raw juice inside. We, making good use of that property of sugar plants, divide the simulation models in a similar way than in real plants. We have considered six different stages in the sugar process, corresponding to diffusion, depuration, evaporation, sugarhouse, steam boilers and pulp dryer. We have developed individual simulation models for each section fixing as boundaries the values of the variables coming from other sections. When we join all sections in the full plant simulations the variables coming from other sections overwrite the boundaries of the individual sections.

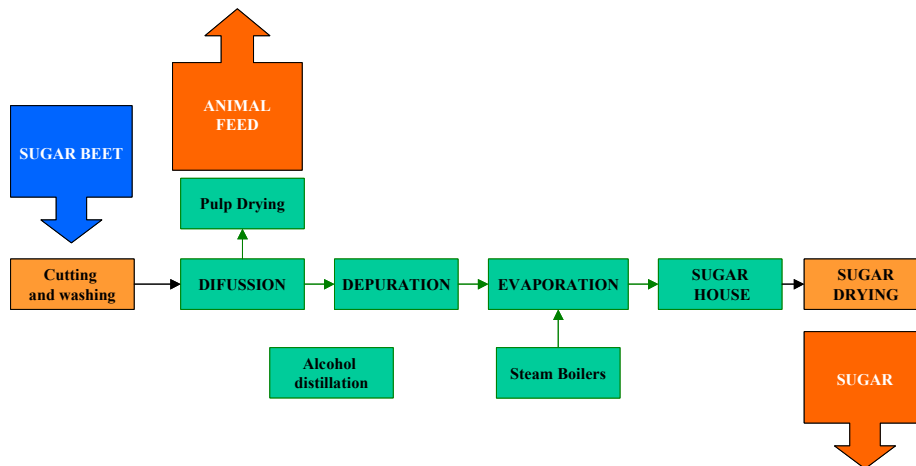


Figure 5. Simplified scheme of a sugar factory

Applying the explained methodology to the sugar factories simulator for the creation of a distributed simulation we have to follow the next steps:

- First, to develop the C++ simulation models, using a library of models of the process units implemented in an object oriented modelling tool for hybrid systems named EcosimPro and to use the tool *CREAOPC* (Figure 6) to create and configure the OPC servers for the previous developed models.

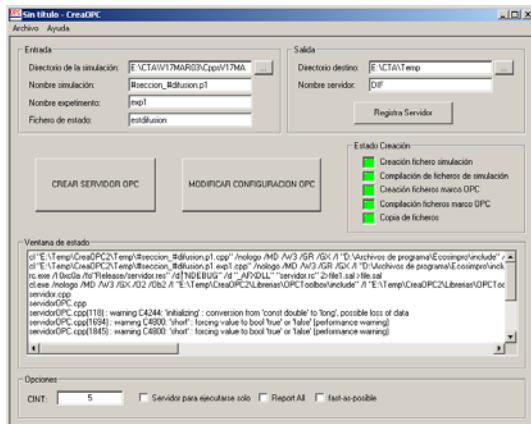


Figure 6. CREAOPC Interface

- Second, to register the OPC servers and give the necessary permissions in DCOM configuration.
- Third, using the *UNESIMCNFG* application to assign the interchanged variables and nodes in which the simulation servers are going to be executed (Figure 7).

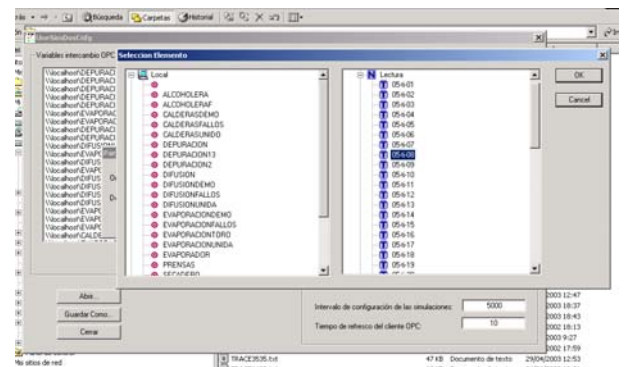


Figure 7. UNESIMCNFG Interface

- Finally, using the *UNESIM* application to run the distributed simulation (Figure 8).

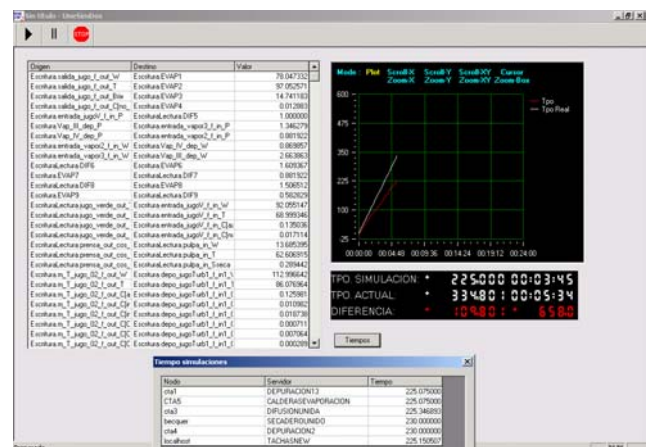


Figure 8. UNESIM Interface

Once created the distributed simulation, following the shown methodology, it can be connected to SCADA system. In the CTA a configurable SCADA system has been developed in order to use it as operator console, so that a operator can manage the simulation as if he was managing the real system.

This SCADA allows the access to the simulations and show data to the operator with the same interfaces that have in the real factory (Figure 9).

In this image we can see the interface of the evaporation section of the simulator

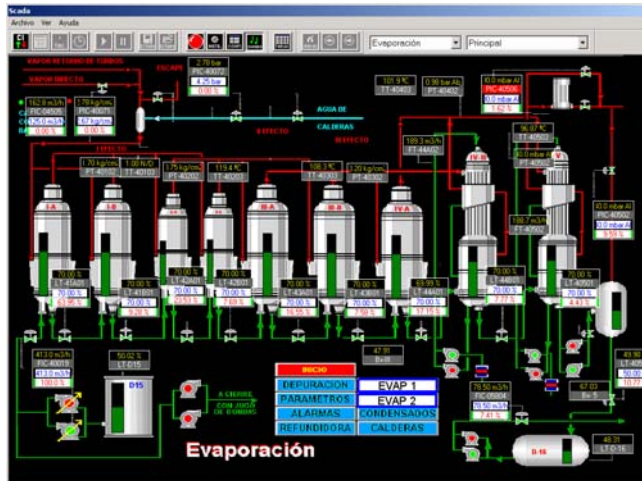


Figure 9. SCADA Interface

This interface is exactly the same than the operators have in the real plant.

Besides this schematic representations, the operators are going to be able to accede to different representation types. In this representations operators can see historical data (Figure 10), data in the present time and allows the operators to manage the process.

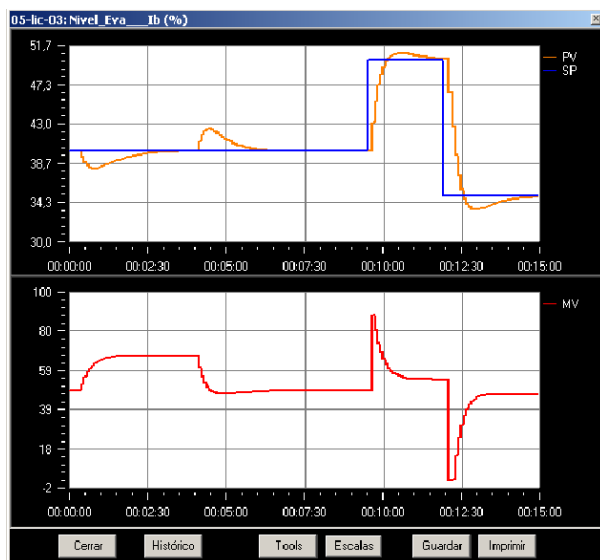


Figure 10. SCADA graphics

CONCLUSIONS

In this article it has been show a methodology for the development of distributed simulations from simulations generated with EcosimPro, communicated via OPC, and their application to the practical case of a sugar factory. In order to facilitate the development, a series of applications has been created that automate the process and allow the execution of the simulation in distributed environments. In the case of study the adopted solution is valid, since the communications time is minimum with respect to the calculation time. Number of interchanged variables is reduced to a hundred of variables every 5 seconds, in a dedicated network, with last generation computers, and knowing the OPC data interchange capacity (Al Chisholm, 1998) can be seen that it produces low communication times. The gain in execution time of the simulations is notable, and increases when increasing the degree of distribution, without affects the communications times, under the parameters before mentioned. Finally we have used this methodology in the development of a whole sugar factory simulator. Showing the necessary steps to make the simulator.

At this moment we are trying to improve the simulation engines, because, as you can see in figure 8, sometimes we have some delay in the simulation time respect real time. That happens when the mathematical model has a lot of algebraic loops, structural singularities, events or when the model inputs are dramatically changed.

REFERENCES

- Al Chisholm, 1998. *DCOM, OPC and Performance Issues*. Intellution Inc.
- CAPE-OPEN, 1999. "Next Generation Computer-Aided Process Engineering Open Simulation Environment: Public Synthesis & Roadmap".
- Department of Defense. Defense Modeling and Simulation Office, 2002. "High Level Architecture. Run-Time Infrastructure. RTI 1.3-Next Generation Programmer's Guide Version 5".
- EcosimPro by EA Internacional, Dynamic Modeling & Simulation Tool <http://www.ecosimpro.com> (2003)
- Frant Iwainitz and Jürgen Lance. 2002. "OPC-Fundamentals, Implementation and Application", ISBN 3-7785-2883-1.
- Microsoft Corporation. 1996. "DCOM Technical Overview".
- Object Management Group, 1998. "The Common Object Request Broker: Architecture and Specification, 2.2 ed, february 1998".
- Softing. 2002. <http://www.softing.com>
- Sun. 2003. "Java Remote Method Invocation". <http://java.sun.com/products/jdk/rmi>
- Zamarreño J.M., Acebes L.F. and Alves R. 2000. "OPC-based real time simulator: architecture and practical example", In Proceedings id 41st SIMS Simulation Conference (SIMS'2000).

INDUCING PARAMETERS OF A DECISION TREE FOR EXPERT SYSTEM SHELL MCESE BY GENETIC ALGORITHM

I. Bruha and F. Franek

Department of Computing & Software, McMaster University

Hamilton, Ont., Canada, L8S4K1

E-mail: {bruha | franya}@mcmaster.ca

KEYWORDS

Expert system shell, genetic algorithms, rule-based systems, classification, data analysis.

ABSTRACT

There exist various tools for knowledge representation, modelling, and simulation in Artificial Intelligence. We have designed and built a software tool (expert system shell) called *McESE* (*McMaster Expert System Environment*) that processes a set of production (decision) rules of a very general form. Such a production (decision) set can be equivalently symbolized as a decision tree.

In real life, even if the logical structure of a production system (decision tree) is provided, the knowledge engineer may be faced with the lack of knowledge of other important parameters of the tree. For instance, in our system *McESE*, the weights, threshold, and the certainty propagation functions – all of these are a part of the machinery handling the certainty/uncertainty of decisions – have to be designed according to a set of training (representative) events, observations, and examples.

One possible way of deriving these parameters is to employ machine learning (ML) or data mining (DM) algorithms. However, ‘traditional’ algorithms in both fields select immediate (usually local) optimal values – in the context of a whole decision set such algorithms select optimal values for each rule without regard to optimal values for the whole knowledge base. Genetic algorithms comprise a long process of evolution of a large population of objects (chromosomes) before selecting (usually global) optimal values, and so giving a ‘chance’ to weaker, worse objects, that nevertheless may prove to be optimal in the context of the whole knowledge base.

In this methodology case study, we expect that a set of *McESE* decision rules (or more precisely, the topology of a decision tree) is given. The paper discusses a simulation of an application of genetic algorithms to generate

parameters of the given tree that can be then used in the rule-based expert system shell *McESE*.

INTRODUCTION

A builder of an expert system usually employs an expert system shell to design and develop a decision-support expert system for a given problem. We have designed and built a software tool (expert system shell) called *McESE* (*McMaster Expert System Environment*) that processes a set of production (decision) rules of a very general form allowing several means of handling uncertainty (Franek 1989; Franek and Bruha 1989). Note that such a production (decision) set can be equivalently exhibited as a decision tree.

In this study, we expect that the logical structure or the topology of a set of decision rules (a decision tree) is given. Even if this logical structure is provided, particularly in real-world tasks, the designer may be faced with the lack of knowledge of other parameters of the tree. These parameters are usually adjustable values (either discrete or numerical ones) of production rules or other knowledge representation formalisms such as frames. In particular, in our system *McESE*, these are represented by weights and thresholds for terms and the selection of the certainty value propagation functions (CVPF for short) from a predefined set. We use the traditional approach of machine learning (ML) and data mining (DM): we adjust the above parameters according to a set of training (representative) observations (examples). However, we use a different and relatively new approach for the inductive process based on the paradigm of genetic algorithms.

Genetic algorithm (GA) encompasses a long process of evolution of a large population of chromosomes (individuals) before selecting optimal values that have a better chance of being globally optimal compared to the traditional methods. The fundamental idea is simple: chromosomes selected according to an ‘evaluation’ are allowed to crossover so as to produce a ‘slightly

different' new one – the offspring. It is clear that the algorithm performs according to how 'slightly different' and 'evaluation' are defined.

In this paper, we present a simulation of applying GAs to generate/adjust the parameter values of a McESE decision tree.

Section 2 of this paper briefly describes our rule-based expert system shell McESE with emphasis on the form of rules. Section 3 then surveys the structure of GAs. Afterwards, Section 4 introduces the methodology of this project including a case study.

RULE-BASED EXPERT SYSTEM SHELL MCESE

McESE (McMaster Expert System Environment) (Franek 1989; Franek and Bruha 1989) is an interactive environment for design, creation, and execution of backward-or forward-chaining rule-based expert systems. The main objectives of the project focused on two aspects: to provide extensions of regular languages to deal with McESE rule bases and inference with them, and a versatile machinery to deal with uncertainty.

The language extension is facilitated through a set of functions with the native syntax that provide the full functionality required (for instance, in the Common-Lisp extension these are Common-Lisp functions callable both in the interactive or compiled mode, in the C extension, these are C functions callable in any C program).

The versatility of the treatment of uncertainty is facilitated by the design of McESE rules utilizing weights, threshold directives, and CVPF's (*Certainty Value Propagation Function*). The McESE rule has the following syntax:

$$R: T_1 \& T_2 \& \dots \& T_n = F \Rightarrow T$$

T_1, \dots, T_n are the left-hand side terms of the rule R and T is the right-hand side term of the rule R . A term has the form:

$$weight * predicate [op \ cvalue]$$

where *weight* is an explicit certainty value, *predicate* is a possibly negated (by ~ or -) predicate possibly with variables, and *op cvalue* is the threshold directive (*op* can either be >, >=, <, or <=, and *cvalue* is an explicit certainty value). If the weight is omitted it is assumed to be 1 by default. The threshold directive can also be omitted. The certainty values are reals in the range 0..1 .

The value of a term depends on the current value of the predicate for the particular instantiation of its variables; if the threshold directive is used, the value becomes 0 (if the current value of the predicate does not satisfy the

directive), or 1 (if it does). The resulting value of the term is then the value of the predicate modified by the threshold directive and multiplied by the weight.

In McESE in the backward-chaining mode, each rule that has the predicate being evaluated as its right-hand side predicate is eligible to fire. The firing of a McESE rule consists of instantiating the variables of the left-hand side predicates by the instances of the variables of the right-hand side predicate, evaluating all the left-hand side terms and assigning the new certainty value to the predicate of the right-hand side term (for the given instantiation of variables). The value is computed by the CVPF F based on the values of the terms T_1, \dots, T_n . In simplified terms, the certainty of the evaluation of the left-hand side terms determines the certainty of the right-hand side predicate. There are several built-in CVPF's the user can use (*min*, *max*, *average*, *weighted average*), or the user can provide his/her own custom-made CVPF's. This approach allows, for instance, to create expert systems with fuzzy logic, or Bayesian logic, or many others (Jaffer 1990).

Any rule-based expert system must deal with the problem of which of the eligible rules should be 'fired'. This is dealt with by what is commonly referred to as *conflict resolution*. In McESE the problem is slightly different; each rule is fired and it provides an evaluation of the right-hand predicate – and we face the problem which of the evaluation should be used. McESE provides the user with three predefined conflict resolution strategies: *min* (where one of the rules leading to the minimal certainty value is considered fired), *max* (where one of the rules leading to the maximal certainty value is considered fired), and *rand* (a randomly chosen rule is considered fired). The user has the option to use his/her own conflict resolution strategy as well.

GENETIC ALGORITHMS: A SURVEY

The induction of concepts from databases consists of searching usually a large space of possible concept descriptions. There exist several paradigms how to control this search, for instance various statistical methods, logical/symbolic algorithms, neural nets, and the like. However, such traditional algorithms select immediate (usually local) optimal values.

On the other hand, the *genetic algorithms* (GAs) comprise a long process of evolution of a large population of individuals (objects, chromosomes) before selecting optimal values, thus giving a 'chance' to weaker, worse objects. They exhibit two important characteristics: the search is usually global and parallel in nature since a GA processes not just a single individual but a large set (population) of individuals.

Genetic algorithms emulate biological evolution and are utilized in optimization processes. The optimization is performed by processing a population of individuals (chromosomes). A designer of a GA has to provide an evaluation function, called *fitness*, that evaluates any individual. The fitter individual is given a greater chance to participate in forming of the new generation. Given an initial population of individuals, a genetic algorithm proceeds by choosing individuals to become parents and then replacing members of the current population by the new individuals (offsprings) that are modified copies of their parents. This process of reproduction and population replacement continues until a specified stop condition is satisfied or the predefined amount of time is exhausted.

Genetic algorithms exploit several so-called *genetic operators*:

- *Selection* operator chooses individuals (chromosomes) as parents depending on their fitness; the fitter individuals have on average more children (offsprings) than the less fit ones. Selecting the fittest individuals tends to improve the population.
- *Crossover* operator creates offsprings by combining the information involved in the parents.
- *Mutation* causes the offsprings to differ from their parents by introducing a localized change.

Details of the theory of genetic algorithms may be found in several books, e.g. (Goldberg 1989; Holland 1975). There are many papers and projects concerning genetic algorithms and their incorporation into data mining (Bala et al. 1995; De Jong et al. 1993; Giordana and Saitta 1993; Janikow 1993; Turney 1995).

We now briefly describe the performance of the genetic algorithm we have designed and implemented for general purposes, including this project. The foundation for our algorithms is the CN4 learning algorithm (Bruha and Kockova 1994), a significant extension of the well-known algorithm CN2 (Clark and Boswell 1991; Clark and Niblett 1989). For our new learning algorithm (*genetic learner*) *GA-CN4*, we removed the original search section from the inductive algorithm and replaced it by a domain-independent genetic algorithm working with fixed-length chromosomes.

The learning starts with an initial population of individuals (chromosomes) and lets them evolve by combining them by means of genetic operators. More precisely, its high-level logic can be described as follows:

procedure GA

Initialize randomly a new population

Until stop condition is satisfied **do**

1. Select individuals by the tournament selection operator
2. Generate offsprings by the two-point crossover operator
3. Perform the bit mutation
4. Check whether each new individual has the correct value (depending on the type of the task); if not the individual's fitness is set to 0 (i.e., to the worst value)

enddo

Select the fittest individual

If this individual is statistically significant **then**

return it

else return nil

The above algorithm mentions some particular operations used in our GA. Their detailed description can be found e.g. in (Bruha et al. 2000; Goldberg 1989; Holland 1975). More specifically, the generation mode of replacing a population is used. The fitness function is derived from the Laplacian evaluation formula. The default parameter values in our genetic algorithm: size of population is 30, probability of mutation $P_{mut} = 0.002$. The genetic algorithm stops the search when the Laplacian criterion does not improve after 10000 generations.

Our GA also includes a check for statistical significance of the fittest individual. It has to comply with the statistical characteristics of a database which is used for training; the P^2 -statistics is used for this test of conformity. If no fittest individual can be found, or it does not comply with the P^2 -statistic, then *nil* is returned in order to stop further search; the details can be found in (Bruha and Kockova 1994).

CASE STUDY

In our project, an individual is formed by a fixed-length list (array) of the following parameters of the McESE system:

- the *weight* of each term of McESE rule,
- the threshold value *cvalue* of each term,
- the selection of the CVPF of each rule from a predefined set of CVPF's
- the conflict resolution for the entire decision tree.

Since our *GA-CN4* is able to process numerical (continuous) attributes, the above parameters *weight* and *cvalue* can be properly handled. As for the CVPF, it is considered as a discrete attribute with these singular values (as mentioned above): *min*, *max*, *average*, and

weighed average. Similarly, the conflict resolution is treated as a discrete attribute.

Since the list of the above parameters is of the fixed size, we can apply the GA-CN4 algorithm that can process the fixed-length chromosomes (objects) only.

The entire process of deriving the right values of the above parameters (*weights*, *cvalues*, CVPF's, conflict resolution) looks as follows:

1. A dataset of typical (representative) examples for a given task is selected (usually by a knowledge engineer that is to solve a given task).
2. The knowledge engineer (together with a domain expert) induces the set of decision rules, i.e. the topology of the decision tree, without specifying values of the above parameters.
3. The genetic learner GA-CN4 induces the right values of the above parameters by processing the training database.

To illustrate our new methodology of knowledge acquisition we introduce the following case study. We consider a very simple task of heating and mixing three liquids L_1 , L_2 , and L_3 . The first two have to be controlled by their flow and temperature; then they are mixed with L_3 . Thus, we can derive these four rules:

$$\begin{aligned}
 R_1: & w_{11} * F_1 [\geq c_{11}] \ \& \ w_{12} * T_1 [\geq c_{12}] \Rightarrow f_1 \Rightarrow H_1 \\
 R_2: & w_{21} * F_2 [\geq c_{21}] \ \& \ w_{22} * T_2 [\geq c_{22}] \Rightarrow f_2 \Rightarrow H_2 \\
 R_3: & w_{31} * H_1 [\geq c_{31}] \ \& \ w_{32} * F_1 [\geq c_{32}] \ \& \\
 & w_{33} * H_2 [\geq c_{33}] \ \& \ w_{34} * F_3 [\geq c_{34}] \Rightarrow f_3 \Rightarrow A_1 \\
 R_4: & w_{41} * H_2 [\geq c_{41}] \ \& \ w_{42} * F_2 [\geq c_{42}] \ \& \\
 & w_{43} * H_1 [\geq c_{43}] \ \& \ w_{44} * F_3 [\geq c_{44}] \Rightarrow f_4 \Rightarrow A_2
 \end{aligned}$$

Here F_i is the flow of L_i , T_i its temperature, H_i the resulting mix, A_i the adjusted mix, $i = 1, 2$ (or 3). The corresponding decision tree is on Fig. 1.

We assume that the above topology of the decision tree (without the right values of its parameters) was derived by the knowledge engineer. The unknown parameters w_{ij} , c_{ij} , f_i , including the conflict resolution then form a chromosome (individual) of length 29 attributes. The global optimal value of this chromosome is then induced by the genetic algorithm GA-CN4.

ANALYSIS AND FUTURE RESEARCH

The primary aim of this project was to design a new methodology for inducing parameters for an expert system under the condition that the topology (the decision tree) is known. We have selected domain-independent genetic algorithm that searches for a global optimizing parameters values.

Our analysis of the methodology indicates that it is quite a viable one. The traditional algorithms explore a small number of hypotheses at a time, whereas the genetic algorithm carries out a parallel search within a robust population. The only disadvantage our study found concerns the time complexity. Our genetic learner is about 20 times slower than the traditional machine learning algorithms.

In the near future, we are going to implement the entire system discussed here and compare it with other inductive data mining tools. The McESE system will thus comprise another tool for rule-base knowledge processing (besides neural net and Petri nets) (Franek and Bruha 1989).

The algorithm GA-CN4 h is written in C and runs under both Unix and Windows. The McESE system has been implemented both in C and Lisp.

REFERENCES

- Bala, J. et al. 1995. "Hybrid learning using genetic algorithms and decision trees for pattern classification", *Proc. IJCAI-95*, 719-724.
- Bruha, I. and S. Kockova. 1994. "A support for decision making: Cost-sensitive learning system", *Artificial Intelligence in Medicine*, 6, 67-82.
- Bruha, I., P. Kralik, and P. Berka. 2000. "Genetic learner: Discretization and fuzzification of numerical attributes", *Intelligent Data Analysis J.*, 4, 445-460.
- Clark, P. and R. Boswell. 1991. "Rule induction with CN2: Some recent improvements", *EWSL-91*, Porto, Springer-Verlag, 151-163.
- Clark, P. and T. Niblett. 1989. "The CN2 induction algorithm", *Machine Learning*, 3, 261-283.
- De Jong, K.A., W.M. Spears, D.F. Gordon, "Using genetic algorithms for concept learning". *Machine Learning*, 13, Kluwer Academic Publ., 161-188.
- Franek, F. 1989. "McESE-FranzLISP: McMaster Expert System Extension of FranzLisp", In: *Computing and Information*, North-Holland.
- Franek, F. and I. Bruha. 1989. "An environment for extending conventional programming languages to build expert system applications", *Proc. IASTED Conf. Expert Systems*, Zurich.
- Goldberg, D.E. 1989. *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley.
- Giordana, A. and L. Saitta. 1993. REGAL: An integrated system for learning relations using genetic algorithms, *Proc. 2nd International Workshop Multistrategy Learning*, 234-249.
- Holland, J. 1975. *Adaptation in natural and artificial systems*, Univ. of Michigan Press, Ann Arbor.

Janikow C.Z. 1993. "A knowledge-intensive genetic algorithm for supervised learning", *Machine Learning*, 5, Kluwer Academic Publ., 189-228.

Jaffer, Z. 1990. "Different treatments of uncertainty in McESE", MSc. Thesis, Dept Computer Science &

Systems, McMaster University.

Turney, P.D. 1995. "Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm", *J. Artificial Intelligence Research*.

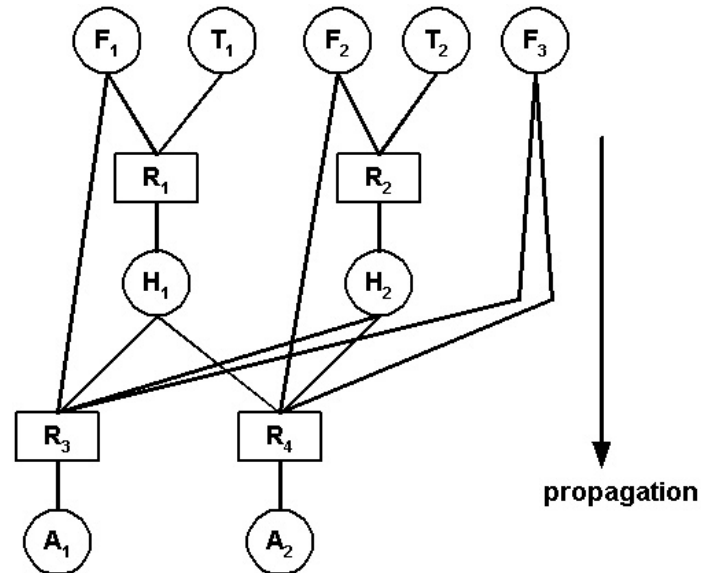


Figure 1: The decision tree of our case study

THE THERMOPTIM SOFTWARE PACKAGE

Nicolas BUS
Centre Scientifique et Technique du Bâtiment
290, route lucioles –BP 209
F-06904 Sophia-Antipolis
E-mail: Nicolas.bus@cstb.fr

Renaud GICQUEL
Ecole de Mines de Paris
60, Boulevard Saint Michel
75272 PARIS cedex 06
E-mail : renaud.gicquel@ensmp.fr

KEYWORDS

Thermodynamics, industrial engineering, optimization, pinch analysis, model analysis, model design.

ABSTRACT

The ThermoOptim software package (www.thermoOptim.com) is mainly a very original and without equivalent systemic environment for modeling energy technologies, which makes possible the adoption of new methods particularly fertile as regards teaching, modeling and optimization.

By assembling models of preset components, it makes it possible to very easily represent various energy systems, from the simplest to the most complex ones. According to cases, the models of these components can be either purely phenomenological, or more technological, i.e. able to carry out sizings or to simulate operation in off-design mode. ThermoOptim differs from other software by highlighting the couplings and interactions between these components, i.e. the systemic properties of the plants studied. It is thus complementary to specialized tools developed by the manufacturers for the development and the manufacture of their components.

The libraries and the methods of calculation of its core enable it to very largely cover the field of energy systems (vapor cycles, refrigeration cycles, combustion, wet mixtures...), and an extension mechanism confers to it great possibilities in term of interoperation: addition of external components, customized control of its calculation scheme.

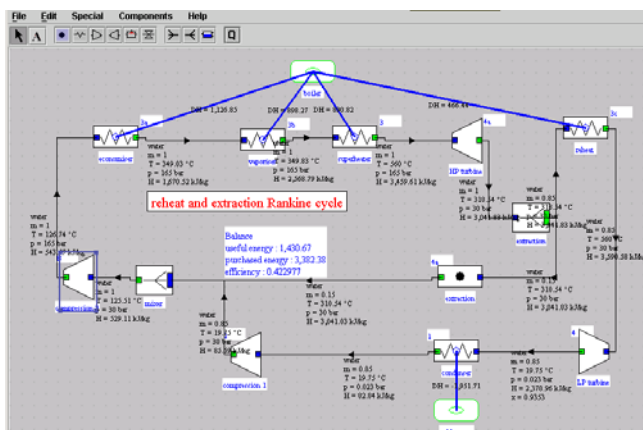


Figure 1: ThermoOptim system modeler

IMPROVING ENERGY SYSTEMS TEACHING

Regarded up to now as particularly difficult by generations of students and engineers, thermodynamics applied to energy systems can today make use of an original and extremely simplified method of training, based on the use of ThermoOptim, in which, to study energy technologies, the accent is put more on the systems and qualitative aspects than on the detailed analysis of the behaviour of the components and the quantification of their performances.

Research into applied sciences didactics shows that the teaching of a discipline is anchored in a double reference frame corresponding on the one hand to certified scientific knowledge such as that produced by the research laboratories (one often speaks about the knowledge or declaratory knowledge), and on the other hand to knowledge relating to the techniques and their applications in the the real world (one speaks then about know-how or procedural knowledge). The first is rather abstract and deal with the theory, while the second is more practical and directed towards action. To make a success of their training, it is more and more generally recognized that students must build their knowledge starting from their own former assets, navigating between these two reference frames, modelling being one of the privileged ways to do it.

For energy systems, it is unfortunately almost always impossible to find models at the same time simple and precise. While hardly caricaturing, one could say that the traditional presentations of the discipline are confronted with a dilemma, the models they lead rely upon either unrealistic, or incalculable. This explains why thermodynamics is often regarded as a matter difficult to teach. The problem has been identified for a long time, and many efforts were made to cure it, but until recently one still missed solutions, and this in spite of the efforts made by the teachers and the evolutions of the programs.

ThermoOptim facilitates the study of energy systems by simplifying the modelling approach and making it more safe. It makes it possible to represent many energy technologies by simple graphic assembly of preset components. Without writing a single line of code, it is possible to establish phenomenological models to calculate their performances with a very high degree of accuracy and to plot their cycles in various thermodynamic charts. Discharged from the usual calculative difficulties on the matter, the students get much more motivated for the discipline and assimilate it much better.

Tested for a few years in the various teaching cycles of the Ecole des Mines de Paris, and used at the end of 2002 in

around fifty higher education establishments (preparatory classes for the Grandes Ecoles, IUT, Universities and engineering schools, in France and abroad), this pedagogy has come to supplement the traditional approach by proposing to the students to initiate themselves to the energy systems study while exploring or while working out by themselves the models of the main energy conversion technologies, this thanks to the Thermoptim software package. They thus have a virtual experimental platform enabling them to work out models of many energy systems. On the basis of experience gained these last years, one may summarize in three main categories the possible teaching approaches that a tool like Thermoptim enables: guided exploration activities, in which the student visualizes models pre-assembled in a catalogue of examples, model assembly activities starting from preset components, in which he/she must himself/herself build and parameterise the system studied, and finally the addition of new components to those of the existing core, which requires of him/her to write the model of the component (choice of the characteristic parameters, the variables of coupling and the set of equations adapted, definition of the graphic interface and computer implementation).

The first category addresses to beginners, especially if they have little time and if they work by themselves, for example while following a e-learning module. The second requires more time, as well as a more thorough training of Thermoptim, in particular on behalf of the teacher, who must be able to correct model construction errors. The third finally is addressed to students already familiar with the software package and sufficiently skilled in thermodynamics to be able to start to design by themselves the component models. They must moreover have a minimum of knowledge in programming with Java or at least C. Interfaces with Fortran can easily be made.

FACILITATING AND MAKING SAFE ENERGY SYSTEMS MODELING

More and more, energy technologies form complex inter-connected systems with multiple functions, these systems themselves being very deeply integrated and having to adapt to evolutionary specifications during their lifespan. To study them, it is desirable to have access to environments allowing to facilitate and make safe their modelling.

Combining systemic and traditional analytical and/or empirical approaches, the Thermoptim software package offers a significant potential in this context. Its graphic environment makes it possible to visually build models of many energy systems, from the simplest ones like a refrigerator to the most complex ones like integrated coal gasification combined cycle power plants bringing into play several hundreds of elements.

Thermoptim enables its users to very easily calculate even complex thermodynamic cycles, and this without having to write any equation or to program a line of code if they use the preset components in the core of the software package. Moreover, thanks to the extension mechanisms under development, they can also customize their tool by defining

substances or components non available in the core of Thermoptim, which are perfectly integrated in the environment.

Not only such a manner of doing notably simplifies the modelling and later on facilitates the use and the maintenance of the model, but especially it makes safe its construction by automating the establishment of the couplings between the various elements which make it up and by guaranteeing their coherence. This last point is essential for the large systems bringing into play several hundreds or thousands of equations whose checking requires to be automated in one way or another.

OPTIMISING COMPLEX ENERGY SYSTEMS THANKS TO EXERGY METHODS

The traditional approaches as regards optimisation of energy technologies, valid for optimising one by one various components of an installation, remain insufficient to guide the designer in the choice of the best complete system architecture. The optimisation of the thermal couplings by extension of the pinch analysis provides a rigorous method for that, while making it possible to optimise the total configuration of the installation, and by guaranteeing the best coherence between the energy requirements and availabilities.

Optimisation by systemic integration, on which we work since ten years, is part of the exergy methods. When one assembles different components to form a system, particular irreversibilities appear in addition of those of each element. These irreversibilities can be qualified as systemic ones because they depend mainly on relative positioning of the components. The reduction of these irreversibilities makes it possible to increase the system effectiveness, thanks in particular to a better internal regeneration. Our work gave place to several publications, collaborations with industrialists (Babcock, EDF, Framatome), and to the implementation in Thermoptim of a set of well validated algorithms.

THERMOPTIM DEVELOPMENTS 2002-2003

In 2002, various improvements of Thermoptim were carried out, the principal ones relating to the following points:

- security of modelling: addition of new tests of coherence and messages, improvement of the editor/simulator connections, in particular for the renaming or the removal of elements;
- introduction of new interactive charts: Clapeyron (P,v) for water and ideal gases, Mollier (h,s) and exergetic (h, xh) or (s, xh) for 6 fluids, and wet mixtures Mollier (w, q');;
- new possibilities for the layout of the cycles: superposition of several cycles on the same chart, plot according to isovalue lines, setting of the plot color.
- In addition, on the basis of experience gained these last years, the reflexion on the various teaching uses of Thermoptim was continued as stated above.

Lastly, a stabilized version of the simulator allowing the study of many thermodynamic cycles is fully operational and documented today (version 1.45). In addition to its teaching applications, it is used in industry (EDF, CEA, Framatome, Elyo, Laborelec...), in particular for its optimization method.

For 2003 the developments are being pursued in several directions:

- 1) at the simulator level:
 - the possibility for a user of customizing the software package by adding to the common core its own substances and/or components, thanks to the addition of external classes dynamically loaded in Thermoptim. This extension results in introducing a distinction relating to the complexity of the couplings which can exist between the components, according to whether they bring into play one or more forms of energy (mechanical and thermal); one speaks thus about mono-functional components (like those available in the core) or multi-functional; various external classes have been implemented allowing to represent components such as fuel cells, diffusers and nozzles for aeronautics motor modelling, chemical Gibbs reactors (CO shift conversion), LiBr-H₂O absorption units;
 - preparation of a version for "technological design", allowing, once a thermodynamic cycle is chosen, to size its various components (surfaces of heat-exchangers, choice of a compressor in a manufacturer catalogue...);
 - extension of this new version to be able to study operation off-design mode of the modelled systems. The existing version is a prototype allowing to study industrial refrigeration machines.
- 2) at the teaching level:
 - the preparation of a e-learning module on energy systems based on the teaching method by guided exploration activities (realization ISIGE/ENSMP);
 - the realization of a first prototype of Intelligent Tutorial System allowing to guide a student while training in applied thermodynamics by using Thermoptim (realization CAOR/ENSMP);
 - the realization, within the framework of the Grande Ecole Virtuelle project of the Group of the Schools of the Mines, of a Web site dedicated to Thermoptim and its teaching uses, which will give access to many resources (realization ISIGE/ENSMP);
 - continuation of exchanges on the use of the Information Technologies for the teaching of thermodynamics,

within Club ALET created in 1999 and gathering today about 25 teachers of the field.

REFERENCES

- R. GICQUEL 2001. Systèmes Energétiques, Tomes 1 et 2, Presses de l'Ecole des Mines de Paris.
- J. TOUSSAINT 1996. Didactique appliquée de la physique-chimie, Nathan Pédagogie, Paris.
- R. GICQUEL, M. WILLIAM, K. AUBER 2001. Optimisation du cycle eau-vapeur d'une centrale IGCC, Conférence hpc'01, Paris.
- B. LEIDE and R. GICQUEL 1997. Systemic approach applied to dual pressure HRSG, International Gas Turbine & Aeroengine Congress and Exhibition, Orlando, Florida.
- B. LINNHOFF 1994. Use pinch analysis to knock down capital costs and emissions, Chemical Engineering Progress.
- CHASE and al 1995. Janaf thermochemical tables, J. Phys. Chem. Ref. Data, Vol 14 Suppl. 1.
- K.K KUO 2000, Principles of combustion, John Wiley & sons.
- D. SPADLING 1979, Combustion and mass transfer, Pergamon Press, Oxford, 1979
- W.M. KAYS and A.L. LONDON 1984, Compact heat Exchangers, Mac Graw Hill

BIOGRAPHY

Renaud GICQUEL is Professor and leads the Energy Studies department of l'Ecole des Mines de Paris. He had worked formerly for the United Nations Conference on New and Renewable Sources of Energy. He has written various books as "Introduction aux problèmes énergétiques globaux" and "Systèmes énergétiques". His research deals with complex thermodynamics systems optimization and he has designed Thermoptim software.

Nicolas BUS has a general engineer degree from l'Ecole Nationale Supérieure des Arts et Métiers de Paris. He had worked for two years as a computer science consultant for various engineering companies designing simulation and information management software. He has worked for 3 years for le Centre Scientifique et Technique du Bâtiment as a software engineer designing commercial software from research results.

Simulation Methodology

DIRECT OPTIMISATION FOR SIMULATION BASED SYSTEM DESIGN

Petter Krus Johan Andersson
Department of Mechanical Engineering, Linköpings Universitet
SE-58183 Linköping, Sweden

KEYWORDS:

Optimisation, simulation, distributed solvers, co-simulation

ABSTRACT

Modelling and simulation is of crucial importance for system design and optimisation. The rapid increase in computational power has now come to a point where complete modelling and simulation of all the sub systems in, for instance, an aircraft or road vehicle is possible.

In this paper it is discussed how direct optimisation based on simulation can be applied to an aircraft flight control system using a flight dynamics model of the aircraft coupled to a model of the actuation system. In this way the system can be optimised for certain flight condition by "test flying" the system. Apart from high efficiency, simulation for use with direct optimisation is also subject to other constraints. They need to be very robust, since parameters (and system dynamics) might have extreme variations, simulation times also needs to be reasonably deterministic. The distributed modelling approach used here is extremely robust and makes it possible to simulate this system much faster than real time on an ordinary PC. This means that system optimisation can be performed in reasonable time.

INTRODUCTION

There are several levels of design from requirement analysis and system architecture down to detail design. There is a clear danger that systems engineering activities are performed only at top level of a design. In order to have an impact on the product development process it must, however, permeate all levels of the design in such a way that a holistic view is maintained through all stages of the design. This can be achieved with common model of the complete system, where the subsystem designs can be tested and optimised in an environment where the interaction with other sub-system and the whole aircraft can be studied. It does, however, impose strong requirements on robustness in the simulation models.

The rapid development in simulation methods and the general increase in hardware performance imply that design methods based on different kinds of numerical optimisation for system design, are becoming much more important. Over the year a number of more or less advanced schemes for design optimization has evolved. There is a relative rich literature in design optimization, see for example G.V. Reklaitis, A.

Ravindran, K.M. Ragsdell, 1983, Papalambros, Douglass, Wilde 1988, or C Onwubiko 2000. There are, however, situations where there is very little information regarding the nature of the object function, gradients can not be obtained explicitly and constraints are implicit. This is true when evaluation of the object function relies on simulation of dynamic systems. In these situations direct search methods are very attractive.

Requirements on simulation

Optimisation based on simulation puts very high demands on the numerical efficiency and robustness of the simulation. Since a high number of simulations need to be done, typically ranging from a few hundred to tens of thousands, low simulation times are of course very important.

The following strategy that has been adopted for the development of the HOPSAN package, developed at Linköping University, for simulation based optimisation:

- Modelling on a detailed equation level using a symbolic math package to generate implementation. In order to provide highly numerically robust models
- The distributed modelling approach for partitioning of systems, based on bi-directional delay lines, see (Auslander 1968) and (Krus et al 1990). In order to be able to handle large systems efficiently.
- Different time step for different parts of the model but constant over time. In order to give short and deterministic simulation times.
- Co-simulation, in order to have an open flexible framework for connecting models from different groups using different simulation tools (Jansson 2001)

There is also ongoing development with the prime objective to introduce a model centric architecture as opposed to the traditional tool-centric architecture. The extensible markup language XML is a prime candidate for defining tools independent model-structures. The *Modelith* schema (<http://hydra.ikp.liu.se/modelith/>) is an effort to define a schema for simulation models. A simulation model can also be a component in a process involving several other design tools.

OPTIMISATION

If a system model in the form of a simulation model is defined, it is possible to use optimisation based on simulation. Using this method, the system is simulated using different sets of system parameters. From each system evaluation a set of system characteristics are obtained, and using these the objective function f is formulated. In general the simulation is used to obtain the performance characteristics of the system and the object function is a function of the system characteristics. A lot of the design effort should be channelled into the formulation of the object function since it determines the design of the final system.

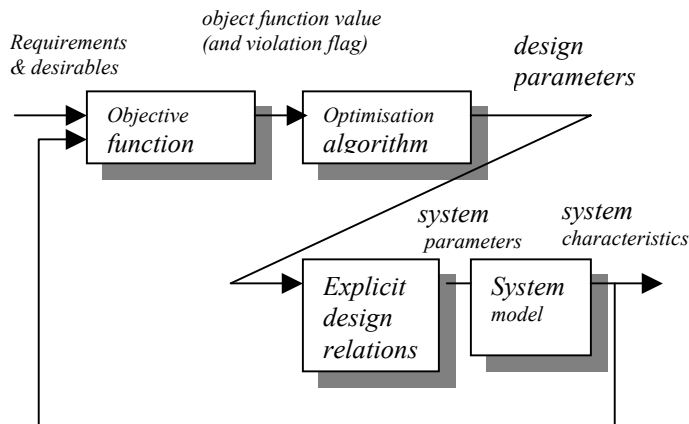


Figure 3. Optimisation based on simulation, with layer of explicit design relations, transforming a few design parameters to many system parameters.

The COMPLEX algorithm

There are basically two families of optimization methods used in engineering. The gradient methods are widely used and are suitable for problems where the gradient of the object function can be calculated explicitly at each point. This is the case in many structure optimization applications. The other group is the non-gradient methods. These methods do not rely explicitly on gradient information in each point. These methods are, therefore, of more general use since gradient information is not generally available, especially if parts of the object function are evaluated using simulation of non-linear systems. The modified version of the original Complex method, by Box 1965) which based on the SIMPLEX method, has been found to be one of the simplest and most easy to use methods, and has been used for system optimization of hydraulic systems See Krus et al 1993. The implementation shown here has also been described in Krus and Gunnarsson 93. This implementation of the Complex method is also used in some major Swedish companies.

EXAMPLE: SIMULTANEOUS OPTIMISATION OF CONTROL SYSTEM AND ACTUATION SYSTEM

As an example an aircraft system was optimised for a manoeuvre. The system model contains a 6-DOF model of the flight dynamics, propulsion, and flight actuation system. There is also a flight control unit. This could either represent an actual flight control unit or just a system needed to represent a pilot to fly the aircraft through the simulation.

There are ten design parameters used for the optimisation in this example. They are

- Size of the aileron pistons
- Size of the elevator and rudder pistons
- The size of the aileron valves
- The size of the elevator and rudder valves
- Gain of the aileron servos
- Gain of the elevator and rudder servos
- FCU gain in pitch
- FCU gain in roll
- FCU gain in yaw
- FCU coupling gain between yaw and roll

The objective function

The main objective is to produce an actuation system that can turn the aircraft as fast as it is possible while having as low weight as possible. That means that the components should have as small size as possible. In addition the pressure variations in the actuators is something that should be kept down in order to promote stable systems. In this example there are no constraint except in the explicit design relations.

Results

The aircraft actuation system was optimised for the case of a sharp 90 deg turn. The simulation started at the time -30 second and the turn was to be initiated at the time 0. The system was then simulated for an additional 40 seconds bringing the total up to 70 seconds. Each simulation run took 4 seconds on a Pentium M 1.6 Ghz processor. The optimum solution was found after 500 simulation runs.

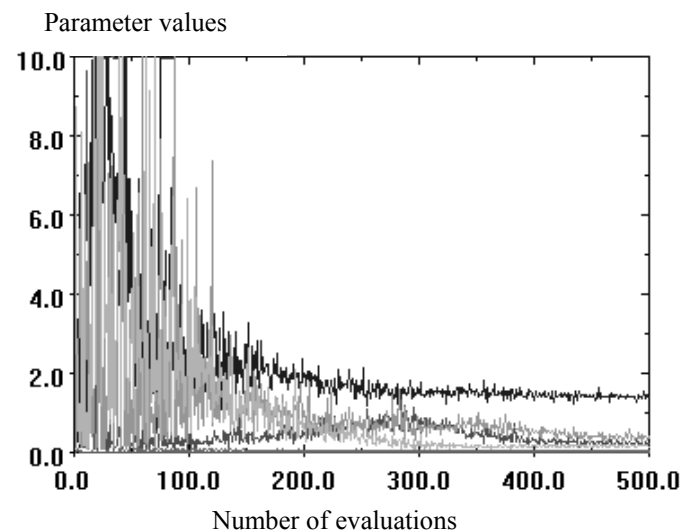


Figure 1. The evolution of design parameters

The optimised flight path is shown in figure 2.

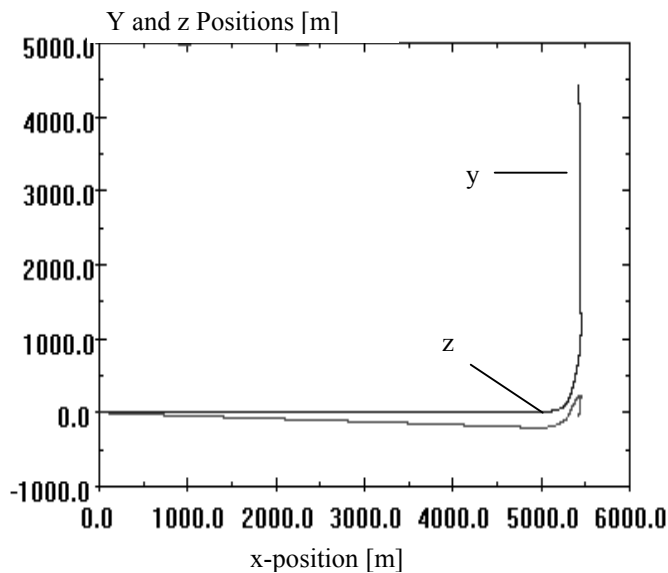


Figure 2. The flight-path of the optimised aircraft.

CONCLUSION

Optimisation techniques are at the core of computational engineering design and in this paper it has been demonstrated how direct-search optimisation methods can be used on full-scale simulation models for system optimisation. The optimisation method found most suited for these kinds of problems is the COMPLEX-RF methods. Simulation based optimisation also puts special requirements on the modelling and simulation methods. Here, distributed solvers with fixed time step has been found most suitable

REFERENCES

- G.V. Reklaitis, A. Ravindran, K.M. Ragsdell. 'Engineering Optimization', John Wiley & Sons Inc. ISBN 0-471-05579-4. 1983.
- P.Y. Papalambros, D.J. Wilde, 'Principles of Optimal Design', Cambridge University Press, ISBN 0-521-42362-7, 1988.
- C. Onwubiko, 'Introduction to Engineering Design Optimization', Prentice Hall, ISBN 0-201-47673-8, 2000.
- D M Auslander, 'Distributed System Simulation with Bilateral Delay-Line Models' *Journal of Basic Engineering*, Trans. ASME p195-p200, June 1968.
- M. J. Box. 'A new method of constrained optimisation and a comparison with other methods'. *Computer Journal*, 8:42-52, 1965.
- P Krus, K Weddfelt, J-O Palmberg, 'Fast Pipeline Models for Simulation of Hydraulic Systems'. Presented at ASME annual winter meet , Atlanta 1991. Also in *ASME Journal of Dynamic Systems, Measurement and Control*. March 1994.
- P Krus, A Jansson, J-O Palmberg, 'Optimization Using Simulation for Aircraft Hydraulic System Design', Proceedings of IMECH International Conference on Aircraft Hydraulics and Systems, London, UK, 1993
- P Krus, A Jansson, J-O Palmberg, K Weddfelt. 'Distributed Simulation of Hydromechanical Systems'. Presented at 'Third Bath International Fluid Power Workshop', Bath, UK 1990.
- J Larsson, 'Concept for Multi-domain simulation with application to construction machinery', Thesis No 870, ISBN 91-7219-955-5, Linköping 2001.

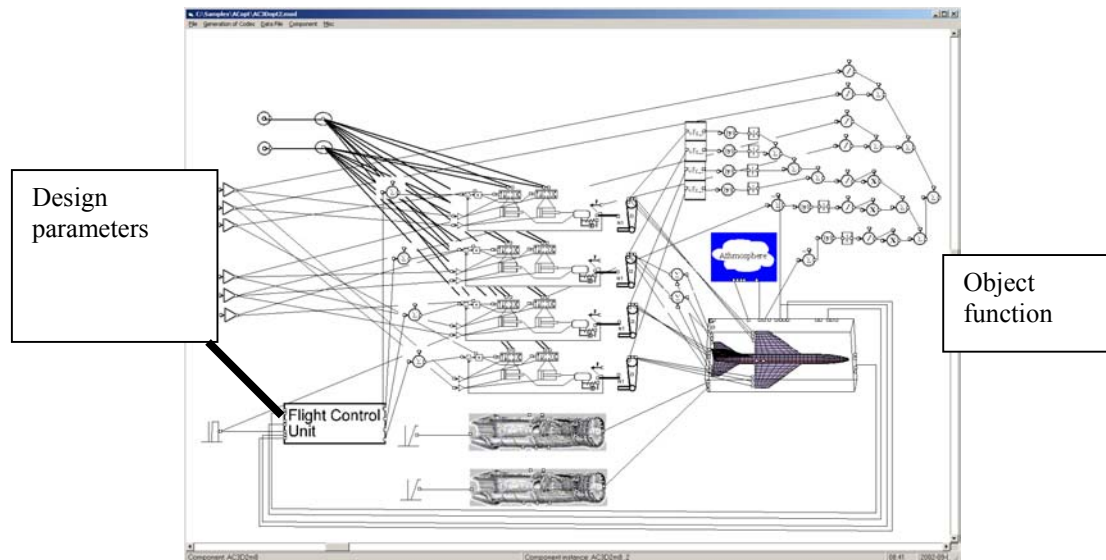


Figure 3. HOPSAN system simulation model of aircraft including flight control actuation system and propulsion. The model also includes arrangements for calculating the object function (upper right) and to transform design parameters into system parameters (left).

APPLYING FORMAL METHODS TO SOFTWARE REUSE

Zina HOUHAMDI
Computer Science Department
University of Biskra
BP 145, Biskra RP, 07000
Algeria
E-mail: z_houhamdi@yahoo.fr

KEYWORDS

Software reuse library, Classification system, Taxonomy, Similarity, Retrieval process, Specification language.

ABSTRACT

Software reuse has been claimed to be one of the most promising approaches to enhance programmer productivity and software quality. One of the problems to be addressed to achieve high software reuse is organizing databases of software experience, in which information on software products and processes is stored and organized to enhance reuse.

This paper presents a new approach to define and construct such databases called the Reuse Description Formalism (*RDF*). The formalism is a generalization of the faceted index approach to classification. Unlike the faceted approach, objects in *RDF* can be described in terms of different sets of faceted and in terms of other object descriptions. This allows a software library to contain different classes of objects, to represent various types of relations among these classes, and to refine classification schemes by adding more detail supporting a growing application domain and reducing the impact of initial domain analysis. In particular, *RDF* provides a specification language based on concepts of set theory capable of representing a rich variety of software and non-software domains; it provides a retrieval mechanism based on exact matches and similarity metrics which can be customized to specific domains; and it provides a mechanism for defining and ensuring certain semantic relations between attribute values.

1. INTRODUCTION

Complex computer programs have placed a growing demand on the talents of software engineers as well as on existing technologies for software development. In order to keep up with the increasing complexity of today's software systems, productivity must be increased and cost reduced in all phases of the software construction process (Barnes and Bollinbger 1991). An important aspect of the projected solution to this growing demand for new software is the development of support technologies to help increase software reuse, that is, the reapplication of knowledge about one system to other similar systems (Anderson and al. 1988, Biggerstaff and Perlis 1989). Rather than starting from scratch in new development efforts, the emphasis must be placed on using already available software assets

(e.g., processes, documents, components, tools). This approach avoids the duplication of work and lowers the overall development cost associated with the construction of new software applications. One important characteristic common to most approaches to software reuse is that they rely, either explicitly or implicitly, on some kind of software repository or library from where the "basic building blocks" are extracted. The fact that software libraries are such an important aspect of most reuse systems, has made software reuse library systems (i.e., systems for designing, building, using, and maintaining software libraries) a very important research topic in the area of software reuse (Houhamdi 2003).

Although these classification models provide the basis for a useful software reuse library system, they have significant limitations and, therefore, can only be regarded as a first step towards a more complete system. They all suffer from one or more of the following problems (Houhamdi and Ghoul 2001):

- Restricted domain. Some reuse library systems have been designed with the purpose of improving reuse at code level. Their representation language usually does not have the expressive power to model more abstract or complex software domain (e.g. software project, defect, or processes).
- Poor retrieval mechanism. One essential characteristic of any software reuse library system is to allow the retrieval of candidate reuse components based on partial or incorrect specifications. This functionality requires the ability to perform similarity-based comparisons, but most systems only provide retrieval based on partial keyword matches or predefined hierarchical structures.
- Not flexible. Software reuse library systems must evolve as the level of expertise in an organization evolves. Because of this, a software reuse library system must be flexible enough to allow the incorporation of new classification schemes or new retrieval patterns, yet this is not the case in most systems.
- No consistency verification. Most software reuse library systems are based on representation models, which must satisfy certain basic predicates for the library to be in a consistent state. Yet, most of these systems do not provide a mechanism for ensuring this consistency.

This paper proposes a classification system for software reuse called the Reuse Description Formalism (*RDF*) which addresses the limitations of current software reuse library systems. *RDF* is based on the principles of *faceted classification*, which have proven to be an effective

mechanism for creating such systems (Prieto-Diaz 1985, Prieto-Diaz 1987). *RDF* is capable of representing a rich variety of software (and no-software) domains; provides a powerful and flexible similarity-based retrieval mechanism; and provides facilities for ensuring the consistency of the libraries.

2. FOUNDATION OF *RDF*

The Reuse Description Formalism uses a generalization of the faceted classification approach proposed by Diaz (Prieto-Diaz 1991) to represent and classify software objects. The faceted index approach relies on a predefined set of *facets* defined by experts. Facets and associated sets of *terms* form a classification scheme for describing components. Component descriptions can be viewed as a records with a fixed number of fields (facets), where each field have a value selected among a finite set of values (terms). Faceted classification scheme has proven to be an effective technique to create libraries of reusable software components. Yet, it suffers from various shortcomings, which limit its usefulness and applicability. The *RDF* approach to classification overcomes these limitations by extending the representation model as follow (Houhamdi 2002):

- Components are replaced by *instances* that belong to several different *classes*. Instances and classes are defined in terms of attributes and other classes, supporting multiple inheritances.
- Facets are replaced by typed *attributes*. Possible types are: integers, string, enumerations, classes, and sets of the above. Having instances as attribute values allows a library designer to create relations among different instances (e.g., that **push** is a component of **stack**).
- The concept of similarity is extended to account for the richer type system, including comparisons of instances of different classes and comparisons of set values.
- Semantic attribute relations can be defined and checked using the **assertion** construct. This facility simplifies the process of maintaining the consistency of the definitions in a software library.
- An integrated language describes attributes, terms, classes, instances, distances, and their dependencies. Descriptions are type checked. The language is based on a formal mathematical model, which makes it both coherent and analyzable.

2.1. Representation model

To understand the representation principles of *RDF*, it is useful to consider descriptions of objects of a particular class as point in a multidimensional space, where each dimension is represented by an *attribute*. Attributes have a name and a list of possible values defined by their associated *type* (i.e., set of values). If a is an attribute name, and v belongs to the a 's type, the assignment " $a = v$ " represent the set all objects whose attribute a is v . Assignment can be combined in expression to define other sets of objects. In particular, if A_1 and A_2 are two assignments, the expression « $A_1 \ \& \ A_2$ » represents the

intersection of the sets A_1 and A_2 . Similarly, " $A_1 \mid A_2$ " represents the union of these sets. In addition, the set of objects that have been defined in terms of a particular attribute a , independently of the value associated with a , is denoted by "**has** a ". The set of objects defined by the "has" operator is a short form of the expression " $a = v_1 \mid a = v_2 \mid \dots \mid a = v_n$ " where the value v_i are the elements of the type of a . A set of objects is called a *class* in *RDF*. Classes can be given a name they are denoted as **class** (E) where E is an expression; i.e., unions and intersections of other sets of objects. If c is a class name, the set of objects it represents is denoted by "**in** c ", and can be combined with other sets of objects in an expression. An object description is called an *instance* in *RDF*. Instances can be given a name and they are denoted as **instance** (E) or $[E]$ where E is an expression. Semantically, an instance must have only one set of attributes, therefore we say that instance (E) is well defined if and only if: (1) E is not a contradiction (i.e., $\text{class}(E) \neq \emptyset$), (2) E defines a mapping from attributes to values, that is, E can be simplified into a consistent conjunct of assignments.

Expressions can also be used to characterize particular sets of instances defined in a *RDF* library. We denote by **set** (E) the set $\{i \mid i \in D \cap \text{class}(E)\}$, where D is the set of instances in the library. In other words, the **set** operator defines the set of instances in the library that belong to the class defined by E .

2.2. Similarity model

The goal of any Reuse Library System is to facilitate the process of finding suitable objects for reuse. *RDF* supports two criteria for selection candidate objects: by exact match and by similarity. For exact matches the construct set (E) already described is used. Similarity-based queries are performed using the construct "query E ", which denoted the list of instances in the library sorted by decreasing similarity to the target object define by E . That is, the first element of the "query E " is the best reuse candidate for $[E]$, the following element the second best, and so on.

As mentioned earlier, similarity is quantified by a non-negative magnitude called similarity distance, which is used as an estimator of the amount of effort required to transform one object into another. Because of this, distances between two object descriptions, A and B , are not symmetric, because the effort to transform A into B is not necessarily the same as the one required to transform B into A . For this reason, whenever a distance is computed, it is important to define which object is the *source* and which the *target*.

Let Z be an object class defined by the set of attributes $Z' = \{A_1, \dots, A_n\}$, and S and T be two instances in this class. Also, let $S' \subseteq Z'$ be the actual set of attributes used to define S , and similarly for T' . The distance from S to T is denoted by $D(S, T)$ and is computed as follows:

$$D(S,T) = \sum_{A \in S' \cap T'} K_A T_A(S.A, T.A) + \sum_{A \in S' - T'} K_A R_A(S.A) + \sum_{A \in T' - S'} K_A C_A(T.A)$$

Where $I.A$ denotes the value of an attribute A on an instance I . The set $S' \cap T'$ represents the attributes shared by S and T , while $S' - T'$ is the set of attributes found in S but not in T , and similarly for $T' - S'$. These three sets are disjoint. In addition, each constant K_A is called the relevance factor of attribute A . Their values fall in the range 0 to 1., and must satisfy the relation $\sum_{A \in Z'} K_A = 1$.

Functions T_A , R_A and C_A are called comparators, and are explained later in this section. The expression for distance $D(S,T)$ is based on the assumption that the overall transformation effort from S to T can be computed using a linear combination of the differences between their respective attributes. In other words, attributes are considered independent of each other when computing similarity. This is a strong assumption that limits the types of domains that can be handled by *RDF*'s similarity model.

Relevance factors.

In general, the distance between two *RDF* objects is given by the sum of the distances between their corresponding attributes. This default scheme gives equal importance to all attributes. In our particular situation, this is not a reasonable assumption. For example, one would consider that the difference between component subsystems is more important than the difference between their numbers of lines of source code. Therefore, the first step required to design comparators is to assign a relevance factor to each attribute in the representation model, that is, to define the amount of influence they have in the computation of similarity distances.

Comparators.

Explained earlier, each attribute has three associated functions T_A , R_A and C_A called comparators. T_A is the transformation comparator and is used to qualify the amount of effort required to transform one value of attribute into another. R_A is the removal comparator and is used to estimate the amount of effort required to eliminate a source attribute value not required in the target specification. Finally, C_A is the construction comparator and estimates the amount of effort required to supply a target value not specified in the source specification. The set of all attribute comparators plus their associated relevance factors define a specific similarity model for reuse library. These functions and values must be specified using a process called domain analysis (Prieto-Diaz 1987) which, among other thing, defines the criteria for similarity for objects in a particular domain. Nonetheless, *RDF* provides default comparators for each type of attribute. These default comparators can be used as a starting point from which to refine the similarity model of a library. This refinement is normally done by assigning attributes non-default comparators using "foreign" functions specified in some conventional programming language.

RDF defines default comparators for each different kind of *RDF* type. Although default comparators are well suited for

certain domains, sometimes it is necessary to define alternative comparators to be able to capture the semantics and relations of specific objects and attributes. For this purpose, *RDF* allows the library designer to define arbitrary comparators, which can be assigned to any attribute or type using the "**distance**" clause.

2.3. RDF Specification language

This section presents a formal definition of the syntax of the *RDF* language. Syntax is presented in a variation of the BNF using the following conventions: Keywords and symbols occurring literally are written in bold; non-terminals are written in italics; *type-name*, *attribute-name*, *instance-name*, *term*, and *class-name* all denote identifiers; symbol, ... means one or more occurrences of symbol, separated by commas; and keyword_{opt} means that the keyword may or may not occur, without affecting the semantics.

Declarations.

A *RDF* library consists of a sequence of declarations. Each declaration either defines a name (of a type, an attribute, an instance, or a class) or describes an assertion that must be true of all instances in the library.

Library ::= *declaration*

Declaration ::= *type-declaration* | *attribute-declaration* | *instance-declaration* | *class-declaration* | *assertion*

Attributes and types.

Software components and other objects are described in terms of their attributes. We can think of attributes as fields of a record describing the object. The declaration of an attribute specifies the type of the values for the attribute. *RDF* supports the following types: number, string, term enumerations, object classes, and homogeneous sets of the above.

Attribute-declaration ::= **attribute** *attribute-name* : *type*;

Type-declaration ::= **type** *type-name* = *type*;

Type ::= *simple-type* | *distance-clause* | **set** *distance-clause* **of** *type*

Simple-type ::= *number* | *string* | {*term*, ...} | *class* | *type-name*

Distance-clause ::= **distance**_{opt} | **no distance** | **distance** {*triplet*, ...} | **distance** * {*triplet*, ...}

triplet ::= *term*_{opt} → *term*_{opt} : *number-literal*

The keyword **distance** by itself is optional and assigns default distance functions. The case "**no distance**" indicates that the distance between values of the associated type is always zero. In the third and the fourth forms of the distance clause, the triplet $t1 \rightarrow t2: n$ means that the distance from $t1$ to term $t2$ is n . if $t1$ is omitted the unspecified value is assumed (i.e., n is creation distance of $t2$). If both $t1$ and the arrow are omitted, the previous $t1$ is assumed. If the keyword distance is followed by the character "*", then the distances between terms not mentioned in a triplet will be set to infinity. If "*" is not

specified, distances between all terms will be adjusted by computing the shortest path between them.

Expressions.

Expression are formed from attribute assignments, the unary operators **has** and **in**, and the binary operators **&** (intersection) and **|** (union).

Expression ::= *attribute-name* = *value* | **has** *attribute-name* | **in** *class-name* | *expression* **&** *expression* | *expression* | *expression* | (*expression*)

The expression “*attribute-name* = *value*” means that the value of *attribute-name* for the instance being defined is *value*. The expression “**in** *class*” means that the instance defined belongs to the *class*; it is similar to a macro-expansion of the expression that defines the class. The expression “**has** *attribute-name*” denotes the condition that the instance being defined has some value for *attribute-name*.

Values.

Values are used in assignment expressions. Values are either simple values or set values. A simple value is either a literal (number or string), a term, an instance, or the value of an attribute of an instance. Set values must denote homogenous sets; they are described either by extension or by intention, using the **set** construct. Only sets of instances can be described by intention.

Value ::= *simple-value* | {*simple-value*, ...} | **set** (*expression*) | **set** (*instance-name* | *expression*)
Simple-value ::= *number* | *string* | *term* | *instance* | **self** | *Instance.attribute-name* | **self.attribute-name**

The construct **set** (*E*) represents the set of all instances in the library that satisfy the expression (i.e., that belong to **class** (*E*)). If the optional *instance-name* is used, the name is bound within *E* to each instance in the library. The dot notation “*instance.attribute-name*” is used to refer to the value of the attribute *attribute-name* of an instance. This notation is similar to that used in other languages to access record fields. The keyword **self** is a reference to the instance defined by the expression in which the value is used. Within an **instance** construct, **self** is bound the instance defined. Within an **assertion**, **self** is bound to every instance in the library in turn. Within nested **instance** construct, **self** is bound to the innermost instance.

Classes.

A class is defined by giving the corresponding expression; the class denotes the set of all objects for which the expression holds. Classes are used to abstract proprieties of instances and also as abbreviations for the corresponding expressions. Classes are also used as types of attributes whose values are instances.

Class-declaration ::= *class-name* = *class*;
Class ::= **class** (*expression*) | *class-name*

Instances.

Instances are defined in terms of an expression. An instance defined by an expression *E* is a representative of the class of instances defined by “**class** (*E*)”

Instance-declaration ::= *instance-name* = *instance*;
Instance ::= **instance** (*expression*) | [*expression*]

An instance may not exist either because the class is empty (i.e., the expression is a contradiction) or because the class is not specific enough (i.e., it defines more than one valid set of attributes) a sketch of a possible simplification and verification algorithm is as follows.

1. Expand all “**in**” propositions with the expressions of the corresponding classes.
2. Transform the expression into disjunctive normal form, as follows:
 - Restructure the expression using associativity laws so that no disjunction occurs within a conjunction.
 - Represent each conjunct as a set of assignments and has propositions.
 - Represent the expression as a set of these conjuncts.
3. For each conjunction do the following:
 - Delete redundant assignments.
 - If there are still two assignments to the same attribute, or there are unsatisfied have propositions, delete the conjunction.
 - Else, delete has propositions (not needed anymore).
4. Delete conjunctions that imply another conjunction.
5. If there no conjunctions left, fail (*E* is a contradiction)
6. If there are more than one conjunction left, fail (*E* is not specific enough)

Assertion.

An assertion specifies a semantic constraint that must be true of all instances in the library. Expressions are used to represent dependencies between attributes, to constrain data types and classes, and to enforce correct typing.

Assertion ::= **assertion** *expression* \Rightarrow *expression*;

The meaning of “**assertion** $E_1 \Rightarrow E_2$ ” is similar to **set** ($E_1 \subseteq \text{set}(E_2)$). This definition does not capture subtleties with respect to the binding of **self**. *RDF* signals false assertions

Queries and distance computations: Queries are used to examine a *RDF* library; they are not part of the library itself. A query command computes a list of instances in the library sorted by decreasing similarity (increasing distance) to the implicit target instance define by an expression. The syntax of queries is:

Query ::= **query** *expression* | **query** *expression* : *identifier*

If specified, *identifier* must be the name of an attribute or a type, and distances are computed using the distance functions associated with the type or the attribute. If *identifier* is not specified, distances are computed using the

default distance functions provided by *RDF*. The distance command is used to compute similarity distances between a pair of values. This command is useful for verifying the definition of distance functions and the results they produce.

$Distance ::= \mathbf{distance} \ source\text{-}value_{opt} \rightarrow target\text{-}value_{opt}$
 $\quad | \mathbf{distance} \ source\text{-}value_{opt} \rightarrow target\text{-}value_{opt} : identifier$

The *source -value* and *target-value* must be values of the same type (e.g., instance names). In case of terms, they must belong to the same enumeration. If both names are specified, the command computes their transformation distance. If only the source value is given, its destruction distance is computed. Finally, if only the target is specified, its construction distance is computed. The *identifier* has the same use as in the case of the query command.

3. CONTRIBUTION OF THIS WORK.

As explain earlier, current software reuse systems based on the faceted index approach to classification suffer from one or more of the following problems: they are applicable to a restricted set of domains; they posses poor retrieval mechanisms; their classification schemes are not extensible; and/or they lack mechanisms for ensuring the consistency of library definitions. The primary contribution of this dissertation is the design and implementation of the Reuse Description Formalism (Houhamdi 2002), which overcomes these problems.

- *RDF* is applicable to a wide range of software and non-software domains. The *RDF* specification language is capable of representing not only software components at the code level, but it is also capable of representing more abstract or complex software entities such as projects, defects, or processes. What is more, these software entities can all be made part of one software library and can be arranged in semantic nets using various types of relations such as "is-a", "component-of", and "members-of".
- *RDF* provides an extensible representation scheme. A software reuse library system must be flexible enough to allow representation schemes to evolve as the needs and level of expertise in an organization increases. The *RDF* specification language provides several alternatives to extend or adjust a taxonomy so as to allow the incorporation of new objects into the library without having to classify all other objects.
- *RDF* has a powerful similarity-based retrieval mechanism. One essential characteristic of any software library system is to allow the retrieval of candidate reuse components based on partial or incorrect specification. *RDF* provides a retrieval mechanism that selects candidate components based on the degree of similarity of their associated library descriptions. This mechanism is based on an alternative refinement process in which components at different levels of granularity can be retrieved. It also includes facilities that allow a library designer to customize the retrieval process by including domain specific function.

- In short, *RDF* addresses the main limitations of current faceted classification systems by extending their representation model.

4. SUMMARY AND FUTURE WORKS

The *RDF* is a general system for creating, using, and maintaining libraries of object descriptions with the purpose of improving reusability in software and non-software organizations. *RDF* overcomes the limitations of the actual systems by extending their representation model and incorporating a retrieval mechanism based on asymmetric similarity distances. In summary, we have presented a software reuse library system called *RDF* and show how its representation model overcome the limitations of current reuse library systems based on faceted representations of objects. Although the *RDF* reuse system has to be an effective reuse tool, its performance and usefulness can be enhanced. Several areas that need more research were identified:

1. *Domain analysis*. In general, to create a library for software reuse it is necessary to perform a domain analysis, the process of identifying, collecting, organizing, analyzing, and representing a domain model and software architecture from the study of existing systems, underlying theory, emerging technology, and development histories within the domain of interest. Domain analysis is currently done by human expert, but several proposals for formalizing and automating this process have been presented in the literature.
2. *Semi-automatic classification*. A method is needed to classify components in terms of a given representation model. In a general, this involves analysis of the different parts of a component (e.g., source code, documentation, etc.), and the use of heuristics to extract attributes based on this analysis.
3. *Similarity distances*. A method is needed to test whether the reuse candidates proposed by the system are truly best ones available in the software library. For example, if we classify a new component A know to be similar to a previously classified component B, we would expect the library system to propose B as a reuse candidate for A. failure to do this could arise due to errors in classification of components A or B, or because of errors in the definition of relevance factors and/or distance comparators.

REFERENCES

- Anderson K.J., Beck R.P., and Buonanno T.E. 1988. "The full computing reviews classification scheme." *Computer review*.
- Barnes B.H. and Bollinbger T.B. 1991. "Making reuse cost-effective." *IEEE Software Engineering*, 13-24.
- Biggerstaff T.J. and Perlis A.J. 1989. "Software reusability, Volume I: Concepts and Models." *ACM Press Frontier Series*, 474-476,.
- Houhamdi Z. and Ghouli S. 2001. "A Reuse Description Formalism." *ACS/IEEE international conference on*

- computer systems and applications*, AICCSA01, Lebanese American University, Beirut, Lebanon.
- Houhamdi Z. 2002. "Describing and Reusing Software Experience." *The international Conference on Computer Science, Software Engineering, Information Technology, e-Business, and Applications*, CSITeA'02. Foz do Iguazu, Brazil.
- Houhamdi Z. 2002. "A Classification Scheme for Software Reuse." *SCS/IEEE 2002. The third Middle East Symposium on Simulation and Modelling*, MESM'2002, Dubai, Emirate united.
- Houhamdi Z. 2003. "Building and Managing Software Reuse Library." *The international Journal of Computing and Informatics, Informatica* (accepted).
- Prieto-Diaz R. 1985. "A Software Classification Scheme." *Ph.D. thesis*, Department of Information and Computer Science, University of California at Irvine.
- Prieto-Diaz R. 1987. "Domain analysis for software reusability." *In proceedings of the 11th international Computer Software and applications Conference*, COMPSA'98. IEEE Computer Society Press.
- Prieto-Diaz R. 1991. "Implementing Faceted Classification for software reuse." *IEEE Transaction on Software Engineering*. 88-97.
- Prieto-Diaz R. and Freeman P. 1987. "Classifying software for reusability." *IEEE Transaction on Software Engineering*, 6-16.

ON THE USE OF CONTROL VARIATES IN THE SIMULATION OF QUEUES WITH HEAVY-TAILED SERVICE

Pablo Jesús Argibay-Losada, Andrés Suárez-González, Cándido López-García,
Raúl Fernando Rodríguez-Rubio, José Carlos López-Ardao and Diego Teijeiro-Ruiz

ETSE de Telecomunicación

Universidade de Vigo, 36200 Vigo, SPAIN

E-mail: Pablo.Argibay@det.uvigo.es, Andres.Suarez@det.uvigo.es, Candido.Lopez@det.uvigo.es,
rrubio@det.uvigo.es, jardao@det.uvigo.es, diego@det.uvigo.es

KEYWORDS

Control variates, heavy tails, queue systems, queue waiting time estimation.

ABSTRACT

Heavy-tailed distributions have been used for modeling several variables involved in telecommunication networks—like page size distributions in Web servers or file size distribution in FTP servers—and in economic models. In G/G/1 queue systems there are few published results to compute their performance parameters analytically, so many times the only method to study them is to use computer simulation. Computer simulation of queue systems where the demanded service time distribution is heavy-tailed presents some difficulties. In this paper we try to gain some insight in those difficulties and evaluate a possible method—based upon control variates—to speed up simulations of G/G/1 queue systems. We present its performance in M/G/1 queue systems, with G heavy-tailed.

INTRODUCTION

Recent research has shown that the simulation of M/G/1 queue systems with G a heavy-tailed distribution presents difficulties such as a slow convergence of the simulation estimated parameters towards their theoretical values [Gross et al, 2002]. In this paper we propose a method to speed up simulations to mitigate the main problem arising from heavy-tailed demanded service time distribution. This method will be integrated in a simulation tool we are developing for accurately analyzing the performance of networks of queues under various traffic patterns.

Random variables with Heavy-tailed distribution may take high values that appear with very low probability but still have important effects on the moments of the distribution, and therefore on some figures of the queue system. Conversely, in light-tailed distributions, the high values appear with so low probability that their effect on the moments of the distribution is often negligible. The mean queue waiting time, \bar{W} , of a M/G/1 queue system depends on the second moment of the demanded service time distribution, \bar{S}^2 (see below). Hence, the heavy-tailed condition of G will decisively influence the value of \bar{W} and the overall performance measures of the system. It seems reasonable

to suppose that the low probability of the high values of S will delay considerably their appearance in the simulation, hence contributing to slow down the speed of convergence towards the theoretical value, \bar{W} , of the estimator. In this paper we will study a M/G/1 queue system with demanded service time distributed as a Pareto random variable. We will estimate its \bar{W} using and evaluating a control variate method in order to reduce the “variance”—it will be infinite in our case study—of the estimator. In a M/G/1 queue system the **Pollaczek-Khintchine** formula for \bar{W} is

$$\bar{W} = \frac{\lambda \cdot \bar{S}^2}{2 \cdot (1 - \rho)}$$

where S is the demanded service time random variable, λ the mean value of the Poissonian arrival process, and ρ the utilization factor of the system [Kleinrock, 1975]. We will use S^2 as a control variate for W and study the rate of convergence of the mean value estimator to its theoretical value. Finally, we will present some conclusions and some further work to try to extend our results to the simulation of G/G/1 queue systems.

HEAVY-TAILED DISTRIBUTIONS

A random variable (RV) X , with cumulative distribution function (cdf) $F(x)$, is said to be heavy-tailed if its complementary distribution function, $1 - F(x)$, has an hyperbolic decaying tail:

$$\exists \alpha > 0 \left| \lim_{x \rightarrow \infty} \frac{1 - F(x)}{x^{-\alpha}} = c \in (0, \infty) \right|$$

The Pareto cdf, clearly heavy-tailed, is given by

$$F(x) = 1 - \left(\frac{m}{x}\right)^\alpha \quad \forall x \geq m > 0$$

where m and α are respectively called the scale and shape parameters. In [Gross et al, 2002] a displaced Pareto distribution with $m = 1$ is used in a M/P/1 queue to show the problems when simulating such system when α is near 2. In this paper we also fix m to 1 to demonstrate the benefits of our method in a similar scenario. The Pareto probability density function (pdf) is given by

$$f(x) = \frac{\alpha \cdot m^\alpha}{x^{\alpha+1}} \quad x > m > 0$$

So the Pareto k^{th} order moment exists if and only if $\alpha > k$. Its mean value exists if and only if $\alpha > 1$ and is given by $\bar{X} = \alpha \cdot m / (\alpha - 1)$. Its second order moment exists if and only if $\alpha > 2$ and is given by $\bar{X}^2 = \alpha \cdot m^2 / (\alpha - 2)$

ESTIMATION OF THE PARETO FIRST AND SECOND MOMENTS

The **Pollaczek-Khinchin** formula shows that \bar{W} depends on the second order moment of S . Hence, it seems reasonable to suppose that any problem concerning the generation of sample values from S will affect \bar{W} . We can give a rough estimation of how well a fixed number of sample values from a Pareto can give good estimations of the second order moment: the weight of the tail of a Pareto beyond a certain limit K , i.e. the probability of getting one value in the range (K, ∞) , is given by

$$\Pr(S > K) = \int_K^\infty \frac{\alpha}{x^{\alpha+1}} dx = \frac{1}{K^\alpha}$$

The probability of getting all the sample values smaller than K in r trials is

$$\left(1 - \frac{1}{K^\alpha}\right)^r \quad (1)$$

In the Pareto case, when α is close to 2, the tail has a great influence in the value of its second order moment.

For example, with $\alpha = 2.1$ the mean value of the corresponding Pareto is $\bar{S} = 1.909$ and the second order moment is $\bar{S}^2 = 21$.

If we generate a sample of 1 million observations, the probability of getting all the values below K , i.e., the probability of having a sample indistinguishable of that from a truncated Pareto with truncation parameter K is

$$P = \left(1 - \frac{1}{K^\alpha}\right)^{10^6} = \left(\left(1 - \frac{1}{K^\alpha}\right)^{K^\alpha}\right)^{\frac{10^6}{K^\alpha}} \simeq e^{-\frac{10^6}{K^\alpha}} \quad (2)$$

Considering the service time RV, S , whose pdf is a Pareto, and a service time RV S_t , whose pdf is a truncated Pareto from the former, with truncation value K , we have

$$f_{S_t}(x) = \frac{f_S(x)}{1 - \Pr(x < K)} \quad x < K$$

with first and second moments

$$\begin{aligned} \bar{S}_t &= \frac{K^\alpha - K}{K^\alpha - 1} \cdot \bar{S} \\ \bar{S}_t^2 &= \frac{K^\alpha - K^2}{K^\alpha - 1} \cdot \bar{S}^2 \end{aligned} \quad (3)$$

From Eq. (2), with $\alpha = 2.1$, with a probability of 99 percent all of the million observations will be lower than 6434. This means that with a probability of 99 percent we are obtaining a sample from a Pareto that is indistinguishable of that from

a truncated Pareto, S_t , with truncation parameter $K = 6434$. This distribution has $\bar{S}_t = 0.999934 \cdot \bar{S}$ but $\bar{S}_t^2 = 0.58 \cdot \bar{S}^2$, so we will probably have an accurate estimation of the mean value of the Pareto, but a very poor one for the second order moment. This will probably make the estimation of \bar{W} too low.

This means that an *a priori* long enough simulation length can be clearly insufficient when the service time distribution is heavy-tailed. To have more accurate results we need many more sample values in the simulation.

For example, if we want the mean value of the generated sample to be approximately 99 percent that of the untruncated Pareto, from Eq. (3) we obtain $K = 10^{20}$. And from Eq. (1) we see that with a probability of 99 percent we will have a truncated Pareto with such truncation value after generating 10^{40} sample values. If we want the truncated Pareto mean value to be a 90 percent of that of the untruncated Pareto, with a probability of 99 percent we will need no less than 10^{19} sample values.

These examples show that although the M/P/1 process, with α near 2, is ergodic in theory, the simulation length to check that ergodicity will be too high to consider the system ergodic in practice. So estimating \bar{W} through the classical estimator (average of the sample values) will be computationally very expensive even if we start our simulations in steady-state, due to the large sample size to be generated before obtaining a representative set for the involved processes.

PROPOSED METHOD

The method we propose will use S^2 as the control variate for W —we know \bar{S}^2 , its theoretical mean value—and will compute each $W_{CV,i}$ for the estimation of \bar{W} as

$$W_{CV,i} = W_i - a \cdot (S_{i-1}^2 - \bar{S}^2) \quad (4)$$

where W_i is the queue waiting time of the i^{th} customer, and S_{i-1} is the demanded service time of the $(i-1)^{\text{th}}$ customer in the system. Actually, we will use the average of a sample of W_{CV} from each simulation.

We choose a as

$$a_{PK} = \frac{\lambda}{2 \cdot (1 - \rho)} \quad (5)$$

the proportionality factor between \bar{W} and \bar{S}^2 in the **Pollaczek-Khinchin** formula, as a study previous to the estimation of its value from the S^2 and W samples, given that in G/G/1 queue systems we do not know a closed form for the theoretical \bar{W} from which we can deduce a value for a .

CLASSICAL SIMULATION AND THEORETICAL VALUE FROM THE MARGINAL PDF OF W

The first alternative method we are going to use for comparison is the classical simulation starting from the empty state, where the underlying physics of the system are simulated,

starting from an empty system and letting it evolve over time, taking into account the new arrivals and the departures of the customers already served. In this type of simulation, we simply estimate \overline{W} from the average value of the queue waiting times of all the customers that have been served in the system, until a certain number of sample values of W is reached in each simulation.

The second method to compare with, generates independent values of W directly from its theoretical marginal pdf, what represents a simulation with the best initialization possible and keeping only one sample value per simulation. In a M/G/1 queue system, we can obtain the marginal pdf of W as

$$f_W(w) = \sum_{k=0}^{\infty} (1-\rho) \cdot \rho^k \cdot f_{r,k}(w) \quad (6)$$

[Benes, 1956] where

$$f_{r,k}(w) = f_r(w) * f_r(w) * \dots * f_r(w) \quad (k\text{-times})$$

the operator $*$ meaning convolution, and $f_r(w)$ being the pdf of the service time residual life.

Eq. (6) represents an infinite sum of individual pdfs. Since each pdf is the convolution of k residual life pdfs, this means that each RV whose pdf is $f_{r,k}(w)$ can be obtained by adding k independent residual lives. We generate one geometric RV to select which $f_{r,k}(w)$ to sample, and the sample value of W will be a sample value of the selected pdf.

So we can generate several samples of size one of W from its marginal pdf and compute their average.

CASE STUDY

In this section we will compare the proposed method with the other two described in the previous section, in a especially difficult case, showing the potential improvement to be achieved in more realistic cases. This would make a simulator implementing the proposed method a more efficient and useful tool.

We will simulate a M/P/1 queue system with $\rho = 0.5$ and a Pareto RV with $\alpha = 2.1$. Using the **Pollaczek-Khinchin** formula we can see that $\overline{W} = 5.5$. Moreover, the variance of W will be infinite. Fig. 1 shows the computed \overline{W} from 100 simulation runs versus the sample size of each one of them for the three specified methods. The bottom line results from 100 classical simulation runs. We see that there is a big difference between the estimated value with sample size 1 million —around 3.1— and the theoretical value —5.5. The line labeled “pdf” shows the computed \overline{W} from 100 sets of independent samples of the marginal pdf of W .

We see that the new estimation of \overline{W} is much better than in the classical simulation due to the fact that we are generating independent sample values of W directly from its marginal pdf, this is, we are in the steady state. But there is still a noticeable difference between the estimated \overline{W} —now around 4.6 for a sample size of 1 million—

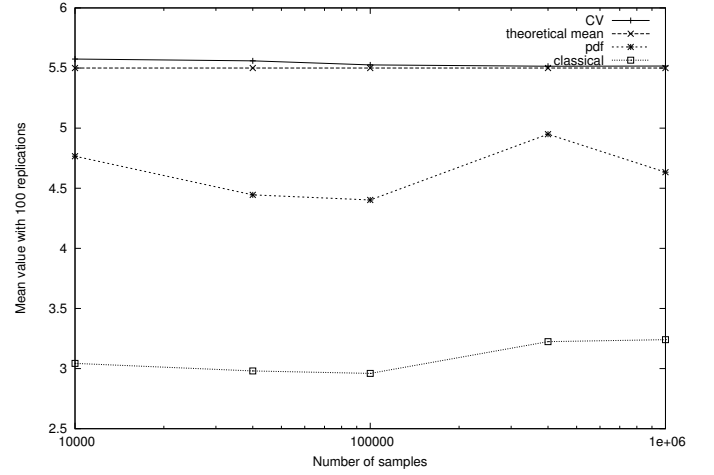


Fig. 1. Rate of convergence to the theoretical value of \overline{W} of the estimated mean values from the classical method, the marginal pdf of W method and the proposed method

and the theoretical value. The line labeled “CV” shows the computed \overline{W} from 100 simulation runs of the proposed method, using the ideal a_{PK} from Eq. (5). We see that the new method produces an average very close to the theoretical mean value, even from the lowest sample length —10000 sample values.

Fig. 2 and Fig. 3 plot the histograms obtained using different simulation sample lengths —10000 and 1 million, respectively— each one for one hundred simulation experiments. In the figures we have represented the theoretical value as a vertical line. The estimated averages of each method are shown in the upper part of the plots. We can see very slow convergence associated to the classical method and the noticeable improvement when we generate independent sample values of W directly from its marginal pdf. We see also that the proposed method gives accurate estimations in all the cases.

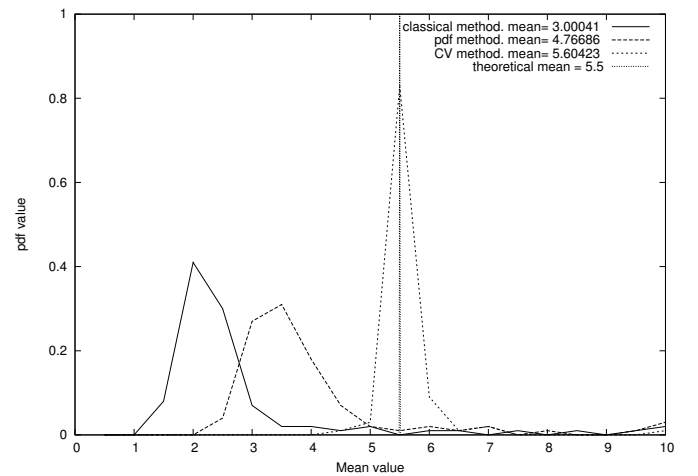


Fig. 2. Empirical histograms of the estimated mean values for \overline{W} with 100 runs of 10000 sample length each, for the classical, marginal pdf of W and proposed methods

Table 1 shows the average and standard deviation for the

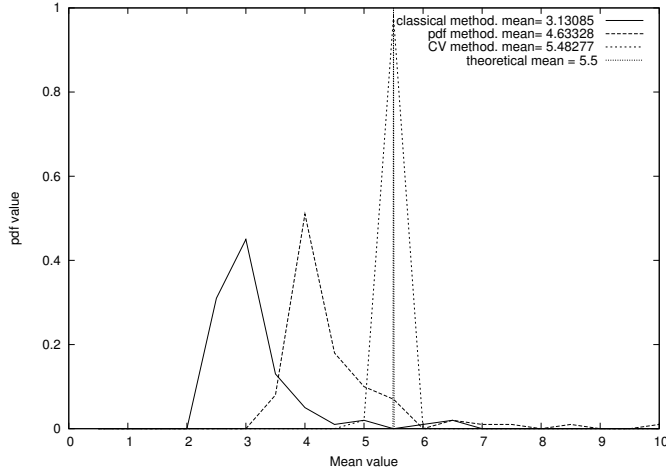


Fig. 3. Empirical histograms of the estimated mean values for \bar{W} with 100 runs of 1 million sample length each, for the classical, marginal pdf of W and proposed methods

classical method, pdf of W method and CV method after 100 simulation experiments with different sample sizes. The standard deviations are calculated using the theoretical $\bar{W} = 5.5$:

$$\widehat{\sigma_W}[n] = \sqrt{\frac{\sum_{i=1}^n (W_i - \bar{W})^2}{n}}$$

TABLE I

COMPARISON BETWEEN THE AVERAGE VALUES AND STANDARD DEVIATIONS FROM THE THEORETICAL \bar{W} FOR THE CLASSICAL, MARGINAL PDF OF W AND CV METHODS.

Samples	10 000		100 000		1 million	
Method	\bar{W}	$\hat{\sigma}[100]$	\bar{W}	$\hat{\sigma}[100]$	\bar{W}	$\hat{\sigma}[100]$
Classical	3.00	4.27	2.92	2.76	3.13	2.48
pdf of W	4.76	7.43	4.40	2.31	4.76	2.52
CV	5.60	0.75	5.52	0.13	5.48	0.077

We can see that the CV method performs well estimating the mean value, and its sample deviation is very low compared to those from the other two methods. Moreover, while we know that the first two methods produce \bar{W} estimators with infinite variance, it seems that the CV one, with known factor a_{PK} , gives rise to a finite variance estimator, presumably tending to zero as the sample size increases. As an indirect proof of the finite variance of \bar{W} , we see in Fig. 6 several 95% confidence intervals for \bar{W} , each one constructed from 100 simulation runs 10000 observations each. The confidence intervals were constructed using the t-Student method, that is, it supposes each W_{CV} average (one simulation sample) to be Gaussian distributed (hence, it supposes its variance finite). The theoretical a_{PK} has been used in the computations, and the achieved coverage is 97 percent, very close to the theoretical one.

In the previous studies we have used $a = a_{PK}$, directly derived from the value in the **Pollaczek-Khinchin** formula—Eq. (5)—, as a previous study before its estimation in G/G/1 systems. We have made 100 runs of different sample

sizes each, with some other values for a , namely $0.5 \cdot a_{PK}$, $0.9 \cdot a_{PK}$, $1.1 \cdot a_{PK}$ and $1.5 \cdot a_{PK}$. We have plotted the estimated \bar{W} versus the sample size in each case in Fig. 4. We see that the greater the difference between the selected a and a_{PK} , the greater the difference in the estimations for samples of 1 million values, and as the simulation sample size grows, the estimations are more accurate. We see in Fig. 4 that for 1 million sample values, the marginal pdf of W method and the CV method with coefficients of $1.5 \cdot a_{PK}$ and $0.5 \cdot a_{PK}$ —50 percent error— give similar accuracy estimators for \bar{W} . In Table 2 we see the computed average and standard deviation values of the estimators of \bar{W} with 100 runs of 10000, 100000 and 1 million sample values each, when we use different values for a other than a_{PK} . We have plotted in Fig. 5 the histograms for 100 runs of 100000 sample values each, for the CV method with a_{PK} , $0.50 \cdot a_{PK}$ and $1.50 \cdot a_{PK}$. All this information will be used to help in the evaluation of different a estimators in further work in order to be able to apply the method to those cases where the queue system is G/G/1 and we do not know \bar{W} in advance.

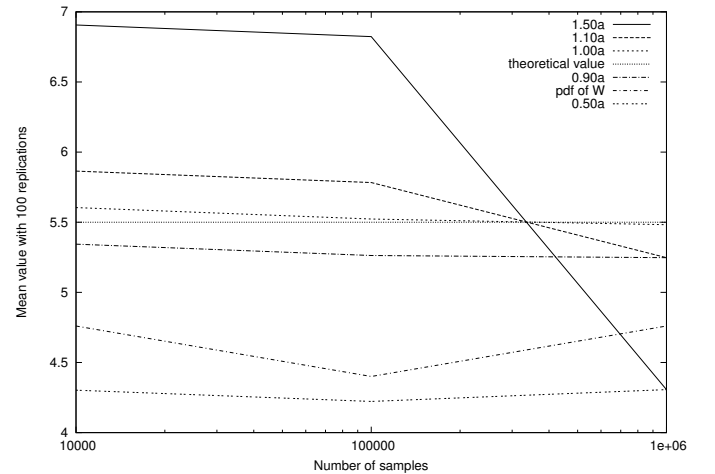


Fig. 4. Average rate of convergence of the proposed method for some different coefficient parameters in the control variate formula

TABLE II

COMPARISON BETWEEN THE AVERAGE VALUES AND STANDARD DEVIATIONS FROM THE THEORETICAL \bar{W} FOR THE CV METHOD WITH VARIOUS VALUES FOR THE CORRECTION FACTOR a .

Samples	10 000		100 000		1 million	
a/a_{PK}	\bar{W}	$\hat{\sigma}[100]$	\bar{W}	$\hat{\sigma}[100]$	\bar{W}	$\hat{\sigma}[100]$
0.50	4.30	2.39	4.22	1.38	4.31	1.24
0.90	5.34	1.01	5.26	0.30	5.25	0.26
1.00	5.60	0.75	5.52	0.13	5.48	0.077
1.10	5.86	0.65	5.78	0.32	5.25	0.26
1.50	6.90	1.66	6.82	1.40	4.30	1.25

CONCLUSION

We see that our method gets accurate estimations of \bar{W} using the information contained in the pairs $(\sum S_i, \sum W_i)$ obtained through the classical simulation. The correlation between this magnitudes plays a very important role in

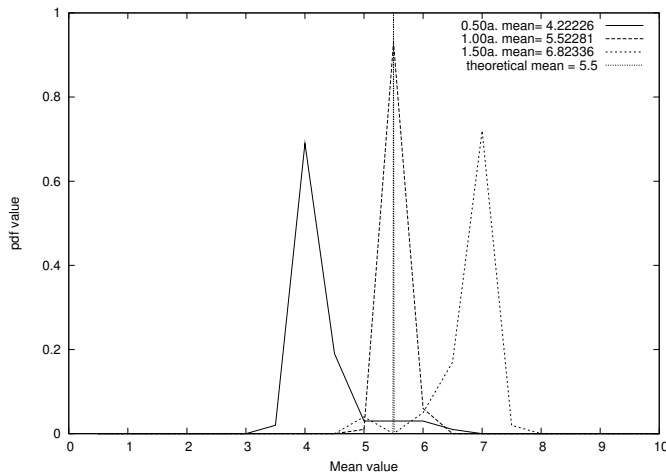


Fig. 5. Empirical histograms of the estimated mean values for \bar{W} with 100 runs of 100000 sample length each, for the CV method with $0.50 \cdot a_{PK}$, a_{PK} and $1.50 \cdot a_{PK}$.

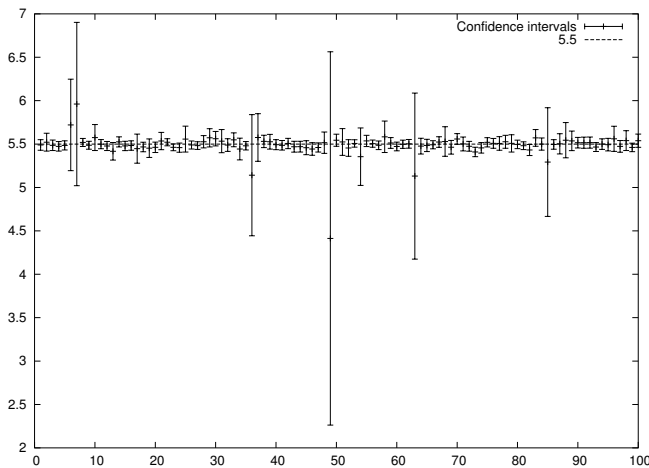


Fig. 6. 95 percent confidence intervals for \bar{W} using the CV method with a_{PK} . The theoretical mean value is 5.5

determining the performance values of the system. So it seems that obtaining independent sample values from the actual marginal pdf of W —to achieve steady state and independence— is not as important as to use the information contained in the relationship between the stochastic processes of interest in the system. We can see that in the **Pollaczek-Khinchin** formula, \bar{W} depends on the values of λ , $\bar{S} - \rho = \lambda \cdot \bar{S}$ and \bar{S}^2 . Since we can get good estimations of λ and \bar{S} —when working with Pareto RVs with α greater than 2—, it seems evident that the difficulties found to estimate \bar{W} have their roots in the difficulties to estimate \bar{S}^2 , and so those are the difficulties found in the classical simulation: the service times generated with a low sample size of S do not give good estimations of \bar{S}^2 . In the proposed method we use the information we have about the demanded service time distribution to help in such estimation. Further work to be done is the estimation of the correction factor from the samples of W and S in order to use the CV method in G/G/1 queue systems, and the evaluation of different methods in order to compute

confidence intervals for the mean value.

REFERENCES

- Gross, D., Shortle, J.F., Fischer, M.J., Masi, D.M.B., 2002. *Difficulties in simulating queues with pareto service*. Proceedings of the 2002 Winter Simulation Conference, 2002.
- Kleinrock, L. 1975. *Queueing systems*. Wiley & Sons.
- Takács, L. 1962. *Single server queue with poisson input simulation*. Operations Research 10:388–397.
- Sigman, K. 1999. *A primer on heavy-tailed distributions*. Queueing Systems. 33: 261-275.
- Benes, V.E. 1956. *On queues with Poisson arrivals*. Annals of Mathematical Statistics. 28, 670-677.
- Bratley, Paul, Fox, Bennet L., Schrage, Linus E. 1987. *A guide to simulation*. Springer-Verlag.

AUTHOR BIOGRAPHIES

PABLO JESÚS ARGIBAY-LOSADA is an assistant professor in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*. He received a telecommunication engineering degree from *Universidade de Vigo* in 2001. Nowadays, he is working toward his Ph.D. in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*.

ANDRÉS SUÁREZ-GONZÁLEZ is an associate professor in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*. He received a Ph.D. degree in telecommunication engineering from *Universidade de Vigo* in 2000. He is a member of ACM. His current research interests include simulation methodology and analysis of stochastic systems.

CÁNDIDO LÓPEZ-GARCÍA is an associate professor in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*. He received a Ph.D. degree in telecommunication engineering from *Universidad Politécnica de Madrid* in 1995.

RAÚL FERNANDO RODRÍGUEZ-RUBIO is an associate professor in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*. He received a Ph.D. degree in telecommunication engineering from *Universidade de Vigo* in 2000. He is a member of ACM. His current research interests include simulation methodology and analysis of stochastic systems.

JOSÉ CARLOS LÓPEZ-ARDAO is an associate professor in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*. He received a Ph.D. degree in telecommunication engineering from *Universidade de Vigo* in 1999. He is a member of ACM.

DIEGO TEIJEIRO-RUIZ is a researcher in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*. He received a telecommunication engineering degree from *Universidade de Vigo* in 2002. Nowadays, he is working toward his Ph.D. in the *Departamento de Enxeñería Telemática* at *Universidade de Vigo*.

ACKNOWLEDGEMENTS

This work was partially supported by the project TIC2000-1126 of the *Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica*.

Compensation for Synthetic Aperture Sonar Motion Errors Using Autofocusing Technique – Simulation Results

Dr. Tahseen Rafik
School of Product and Engineering Design
University of Wales Institute, Cardiff
Western Avenue, Llandaff
Cardiff CF5 2YB, UK
E-mail: trafik@uwic.ac.uk

KEYWORDS

Synthetic Aperture Sonar, Seabed Mapping, Motion Errors, Autofocus.

ABSTRACT

Aperture Synthesis is a technique used to produce high-resolution images using a relatively small sensor fixed on a moving platform. The technique is based on storing successive snapshots of the target's scene as the platform moves then process the collected data as if it has been received from a large physical aperture. This will produce a high-resolution image of the target scene. The technique has been successfully used in astronomy and in both satellite and aircraft borne radar. However, its use in sonar was limited mainly due to the difficulties in maintaining stable track in water, which causes blurring in the images produced.

In this study, the use of Autofocusing technique to reduce the effects of motion errors in Synthetic Aperture Sonar is investigated. Results showed that Autofocus is very effective in removing the effects of motion errors and in restoring the focus of synthetic aperture sonar images.

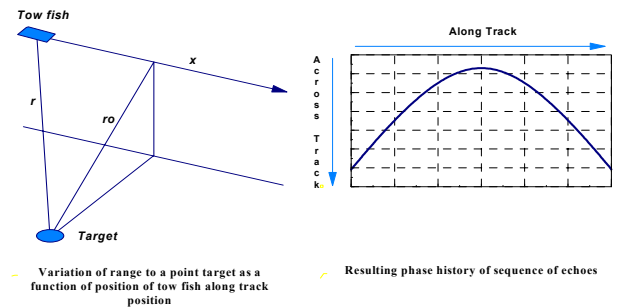
INTRODUCTION

Sidescan sonars are now a widely accepted and important tool of ocean technology. Broadly, they fall into one of two classes: the short-range high-resolution systems working at a relatively high frequency (e.g., about 100kHz). and the long-range low-resolution systems working at 10kHz or less. The former have a wide range of applications in surveying of oil wells, pipelines and shipwrecks, and in defense in mine-hunting. The longer range systems such as GLORIA have contributed significantly to the study of the deep ocean floor. The original method of determining seabed topography is by means of an ship-mounted echo sounder, covering a given area in a raster fashion. This is slow and leaves large undetected areas between the survey lines; multibeam systems have been developed which give improved coverage. An alternative technique, the bathymetric sidescan sonar, was developed in the 1970s, inferring the angle of arrival of an echo from the seabed (and, hence, the seabed topography) from the phase difference measured at two closely-spaced receivers. A review of the historical development of the technique and of its principles of operation has been given by Denbigh (Denbigh, 1983). Several commercial systems have been developed, notably the Bathyscan 300 and SeaMARC systems (Blackington, 1983). The technique has some similarities to radar interferometry, and has the same problem of unambiguous reconstruction of the surface, since the phase is only measured modulo- 2π .

Synthetic aperture techniques have been very successfully applied in radar, both from aircraft and from satellites. Several studies have shown that it is possible to apply synthetic aperture processing to sonar systems (Cutrona, 1975, Williams, 1976, Lee, 1979), and several experimental systems have been built and evaluated. A particular attraction of synthetic aperture systems is that the along-track resolution is independent of range (in principle equal to half the along-track dimension of the transducer array), so high-resolution images can be obtained without the need for a physically large array. A number of problems have been identified, some common to both radar and sonar and some particular to sonar. These have been the subject of research during the past two decades: (i) the stability of the sonar propagation medium, (ii) estimation and correction of motion errors, (iii) the sampling of the synthetic aperture (a consequence of the much lower velocity of propagation in the sonar case), and (iv) the processing required to form the synthetic aperture image. The first three, to some extent, remain and are discussed later in this paper; the fourth has been overtaken by increases in processing power and development of suitable processing architectures.

BASIC PRINCIPLE OF SYNTHETIC APERTURE SONAR (SAS)

Synthetic Aperture is a technique used to produce high-resolution image of target scene by synthesising the effect of very long physical phased array (Figure 1).



$$r = [r_o^2 + x^2]^{1/2} \quad \varphi(x) = -\frac{2\pi}{\lambda} \cdot 2r$$

Figure 1: Synthetic Aperture Sonar

The principle of aperture synthesis consists of storing successive echoes obtained from a moving platform (usually a tow fish), and subsequently synthesising the effect of a large along-track phased array by correcting the phase excursions of echoes in a given direction and summing the sequence of echoes, hence providing high along-track (cross range) resolution.

MOTION ERRORS

The theory of SAS relies heavily on maintaining coherence across the length of the aperture being synthesised. Coherence can be reduced by the phase errors due to unexpected signal path introduced by unknown motions between the target and the towed array. The distorted geometry can be divided into two categories: along-track and across-track. It has been shown (Meng, 1995, Checketts, 1986) that across-track acceleration and along-track velocity errors are the most critical motions in synthetic aperture imagery as they cause image defocusing. These motion errors can be compensated by internal navigation information, but such information is not always available or not precise enough to produce acceptable results. At the same time, there has been significant research work on trying to correct for these motion errors using signal and image processing techniques. One of the most promising signal processing technique to solve the SAS motion error is Autofocusing (Finley, 1995).

Figure 2 shows the six types of platform motion error, which represent three rotational and three translation motion errors. The rotations mainly affect the beam orientation and hence the echo amplitude, while the translations mainly disturb the aperture sampling and hence cause distortion in the image (Checketts, 1986).

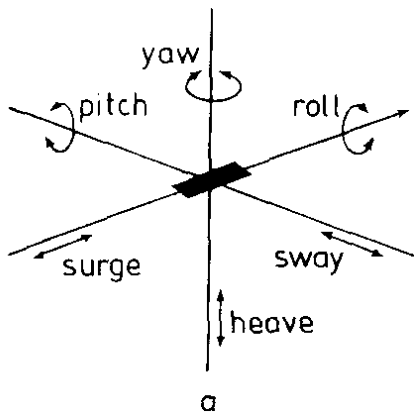


Figure 2: Motion Errors in SAS.

MOTION COMPENSATION

It is well known that motion errors cause distortion (blurring) of synthetic aperture images. The errors can be estimated and compensated either by inertial navigation techniques, or using the image data itself by means of so-called autofocus methods. Of key importance is the relationship between the spatial scale (i.e. wavelength) of the error and the length of the synthetic aperture. In the following Section we review various autofocus techniques and comment on their potential to estimate and correct for errors in interferometry

AUTOFOCUS TECHNIQUES

Autofocus techniques rely on maximising some parameter of the image itself. Four common techniques are power spectra estimation, contrast optimisation, multi-look registration, and phase gradient autofocus (Blake, 1995). The approach adopted in each of these techniques is described briefly below:

(i) *Power spectra Estimation:* This technique uses the received signal from a single point scatterer, which will itself provide the phase history for a target with the towfish path. The matched filter can then be directly derived from this. However, in practice, if the image is badly focused then the identification of a single scatterer may not be reliable. Alternatively the spectrum of the received signals may be used to estimate the bandwidth of the received signal (from all scatterers). In this process the phase is lost so the matched filter cannot be directly estimated, but from the extremes of the bandwidths the chirp rate can be determined and the matched filter estimated.

(ii) *Contrast optimisation:* This technique maximises the contrast of the image, the contrast being a statistical measure of the image. The contrast of a highly structured part of the image is maximised and then this is similarly applied to the surrounding image, or all of the image. This technique is described in more detail below.

(iii) *Multilook image registration:* Multilook registration divides the synthetic aperture into a number of subapertures, forming a separate image or 'look' with each. The looks are coregistered by cross-correlation, which yields an estimate of the motion errors. A disadvantage of this technique is that each of the image strips formed inevitably have a lower resolution due to the shorter aperture used for each image strip.

(iv) *Phase gradient autofocus:* This works by identifying strong targets in the distorted image as a function of range, employing a circular-shifting process to center the targets in azimuth, windowing, and then integrating to estimate the actual parabolic sensor-target path, relying on the facts that the speckle will average to zero and the actual path is only a weak function of range. The algorithm is claimed to work even with images of low contrast.

All the above techniques rely on post processing an image containing at least one highly structural target which allows a reference for the autofocusing as opposed to background scatter. For this reason it may be necessary to deliberately place transponders within the target scene to allow proper operation of the auto-focus algorithms. Also (particularly for the contrast optimisation and multilook registration algorithms), they will work best in correcting relatively low frequency errors, that is, of the same order as the synthetic aperture length. This suggests that the autofocus techniques may be complementary to the inertial navigation methods, the former correcting low-frequency errors and the latter correcting high frequency errors. The contrast optimisation

autofocusing method was implemented and its application is now discussed.

CONTRAST OPTIMISATION

For this technique it is assumed that at areas within the image where there is highly structural content (e.g. a single scatterer) the standard deviation of the image can be used as a measure of the focus of the image; the sharper the focus the greater the maximum value of the target point on the image. The measure of contrast is then the standard deviation of the image, usually normalised by dividing by the image mean. The parameter which is adjusted as the independent variable in order to maximise the contrast is the along-track velocity. Evaluating the contrast against along-track velocity enables autofocusing to be implemented to detect effects of both along-track and across-track errors.

The motion errors that affect the focusing of an image are: across-track acceleration, along-track velocity and along-track acceleration. The defocusing effect of across-track acceleration can be related to an equivalent effect produced by along-track velocity, and hence both motion errors can be considered as one: the effective along-track velocity. Along-track acceleration, if not too significant, can be approximated in a piecewise fashion by velocity.

The contrast optimisation algorithm first identifies an image area of highly structural content on which the optimisation will be performed. This is taken to be an area where the signal-to-noise ratio is a maximum. The image area is then re-processed using a set of different along-track velocities in order to maximise the contrast parameter for the image area, as described above. When the optimum velocity has been determined (to the desired resolution) the complete image is reprocessed at this velocity to produce the focused image.

If the motion error varies significantly over the aperture then the optimum velocity evaluated for one area of the image may not be valid for other aperture positions. In this case the image may be divided into strips and the optimum velocity evaluated for a point of structural content within each strip. The velocity will then apply only to that strip. If there is no significant structural content within a strip then an adjacent velocity may be an adequate estimate.

SIMULATION

In order to study the effects of different motion errors on the images of SAS and the effectiveness of Autofocus technique to remove these effects, a simulation platform was developed using Matlab.

Figure 3 below shows the simulation set-up. The target scene S-S' to be observed is 100cm long by 20cm wide with 5 point targets at the same range, 3m away from the sonar track. The sonar platform transverses through a 5m nominal track (dashed line) from D to D'. The actual sonar trajectory is nominally indicated by a solid freeform line. A 2m long synthetic aperture (thick solid line) slide from P to P' to form each pixel in the scene S-S'.

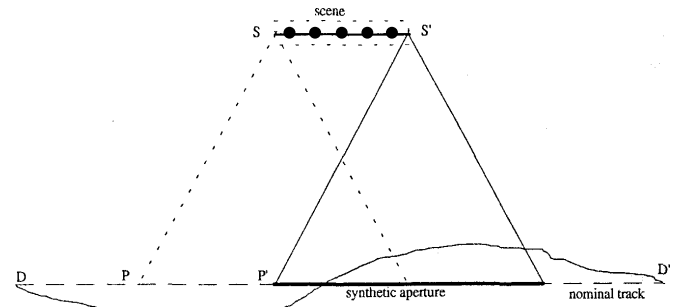


Figure 3: Simulation Geometry

The image of the whole scene was also split into five strips in order to let autofocusing cope with the non uniform track errors. The autofocusing processing on each strip used the best autofocus parameter of the previous strip, but always started with a wide triple estimate spacing, i.e. several depths of spacing.

The image formed with error free aperture is shown in Figure 4 below.

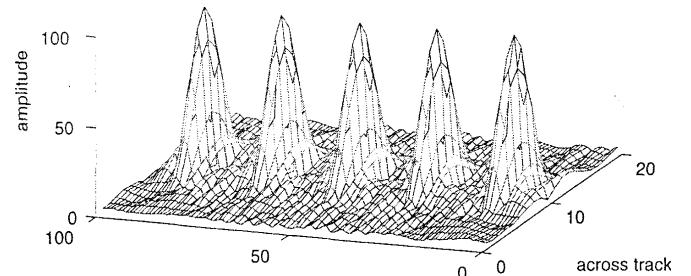


Figure 4: The Error-Free Image

The fairly simple examples for the sonar trajectory with motion errors is that the platform traversed along the straight line with some non-uniform sampling spacing errors. In the following, some simulations for non-uniform sampling spacing errors are presented. Figures 5a, 6a and 7a show the defocused images caused by sampling spacing errors while Figures 5b, 6b and 7b show the same images using autofocus technique.

RESULTS

Along Track Sampling Spacing Errors

Case 1: linearly changing from -8 times depth of spacing to +8 times depth of spacing during sonar moving from D to D'

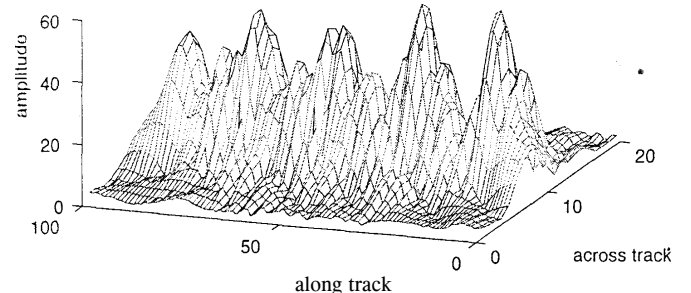


Figure 5a: Image formed from data with spacing error

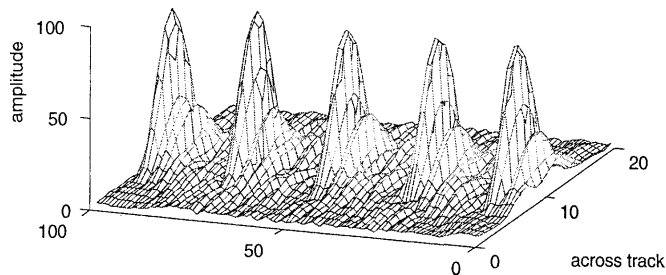


Figure 5b: Image formed with Autofocus

Case 2: Changing in sine form, one cycle within the track D-D' with the amplitude 16 times depth of spacing.

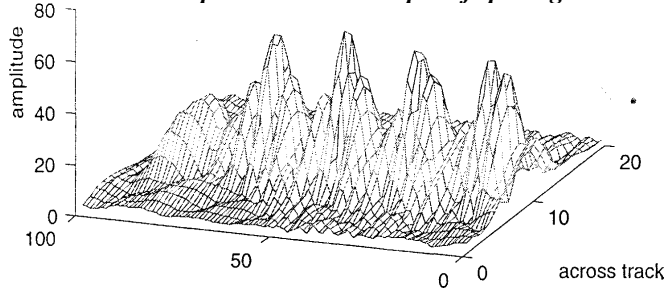


Figure 6a: Image formed from data with spacing error

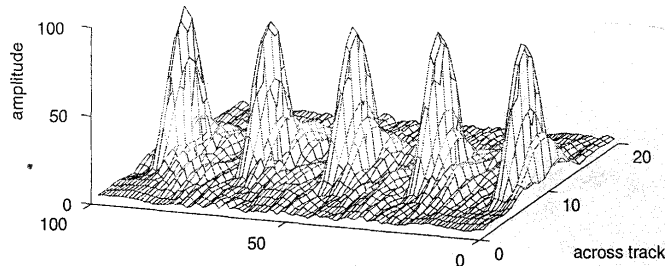


Figure 6b: Image formed from data with Autofocus

Case 3: Changing in sine form, two cycles within the track D-D' with the amplitude 16 times depth of spacing.

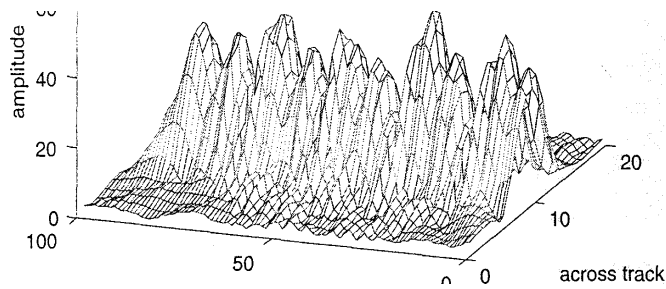


Figure 7a: Image formed from data with spacing error.

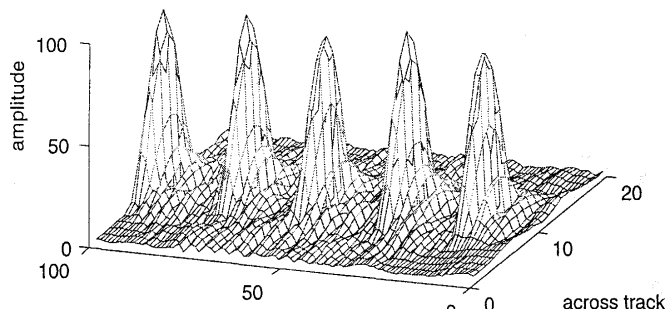


Figure 7b: Image formed from data with Autofocus

Along Track Displacement Errors

For most practical situations, the actual trajectory of the towfish would not be straight but a kind of periodic like curved line, e.g. the error motion caused by a surface wave. Some simulations for such kinds of track with lateral displacement from the nominal track are presented below.

Case 4: Changing in sine form one cycles within the track D-D' with the amplitude 2λ .

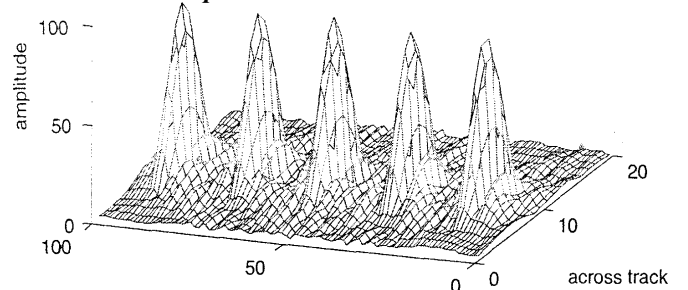


Figure 8a: Image formed from data with track error.

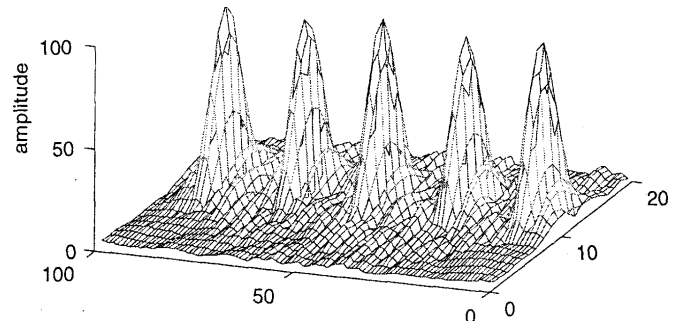


Figure 8b: Image formed from data with Autofocus

Case 5: Changing in sine form five cycles within the track D-D' with the amplitude 2λ .

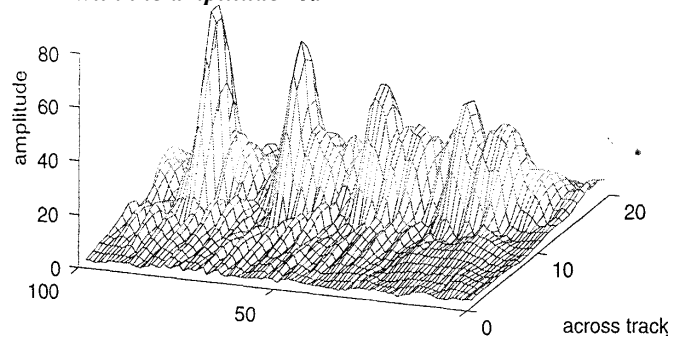


Figure 9a: Image formed with the track errors

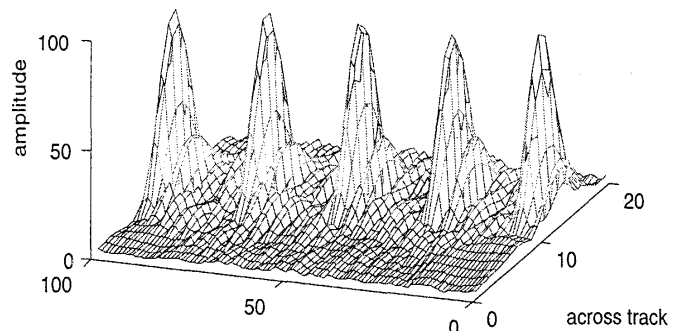


Figure 9b: Autofocused image

Case 6: Changing in sine form ten cycles within the track D-D' with the amplitude 2λ .

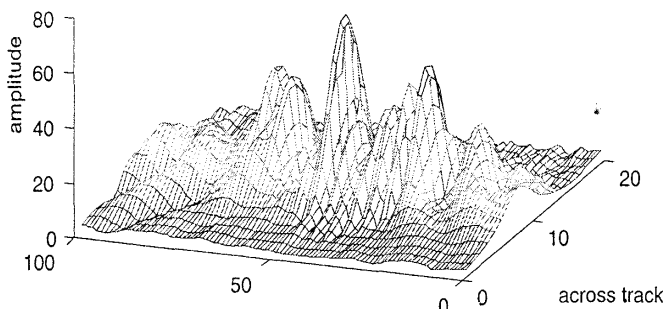


Figure 10a: Image with track error

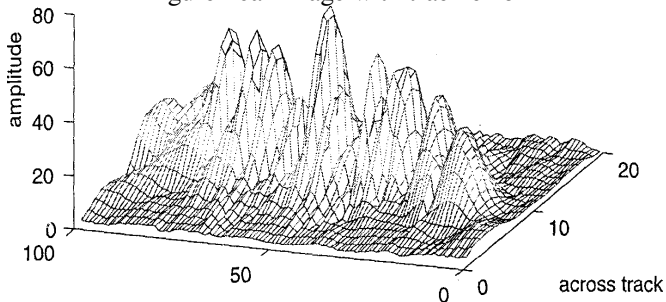


Figure 10b: Autofocused image

Case 7: Random errors within the track D-D' with the amplitude 2λ .

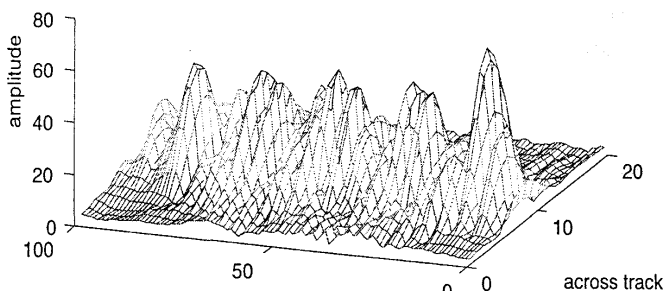


Figure 11a: Image with track error

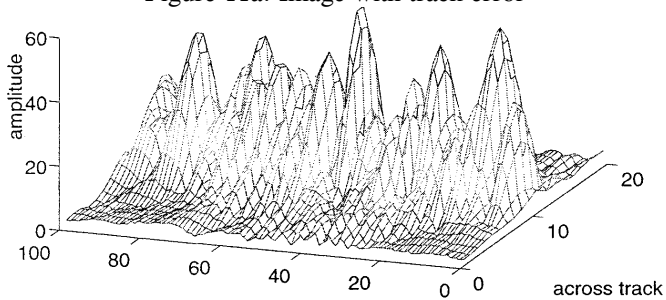


Figure 11b: Autofocused image

DISCUSSION

Along Track Sampling Spacing Errors

In these simulations, the two-cycle in track sine form spacing error caused most severe defocus. This indicated that the most harmful spacing error was the periodic like errors with the wavelength similar to the synthetic aperture length.

Along Track Displacement Errors

Since non-uniform across-track displacement from the nominal track was caused by non-zero across-track acceleration, these track errors can be corrected by compensating for the across-track acceleration. The across-track acceleration can be compensated for by its equivalent along-track velocity error, further by the equivalent along-track sampling spacing error. Therefore, the along-track spacing was still used in the autofocus processing to correct the track error.

In these simulations, the track error with one cycle in track caused the least defocus; therefore, the improvement on the focus of their autofocused image was not significant. However, the image corresponding to the track error of five cycles in track was severely defocused, and the autofocus improved its focus dramatically. For the error with ten cycles in track or the periodic like random error (Case 6 and Case 7), the performances of the autofocus processing were not good. It seemed to agree with the suggestion made by I. P. Finley and J. W. Wood which was that the autofocus couldn't cope with the motion error with a structure much finer than the synthetic aperture.

The results presented in this study demonstrated the good performance of Autofocus algorithms in focusing synthetic aperture sonar images.

REFERENCES

- DENBIGH, P.N.: 'Stereoscopic visualization and contour mapping of the sea bed using a bathymetric sidescan sonar (BASS)'. *Radio & Electron. Eng.* 1983, 53, (7/8), pp. 301-307
- BLACKINGTON, J.G., and HUSSONG, D.M.: 'First results from a combination sidescan sonar and seafloor mapping system (ScaMARC II)'. *Proceedings of the Offshore Technical Conference*. 1983, pp. 307-314.
- CUTRONA, L.I.: 'Comparison of sonar system performance achievable using synthetic aperture techniques with the performance achievable by more conventional means', *J. Acoust. Soc. Am.* Aug. 1975, 58, (2), pp. 336-348
- WILLIAMS, R.E.: 'Creating an acoustic synthetic aperture in the ocean', *J. Acoust. Soc. Am.*, July 1976, 60, (1), pp. 60-73.
- LEE, H.E.: 'Extension of synthetic aperture radar (SAR) technique to undersea applications'. *IEEE J. Ocean. Eng.*, Apr. 1979, 4, (2), pp. 60-63
- MENG, Z.: 'A study on a synthetic aperture sonar'. PhD thesis, Loughborough University, Loughborough 1995.
- CHECKETTS, D.G., and SMITH, B.: 'Analysis of the effects of platform motion errors upon synthetic aperture sonar', *Proc. Inst. Acoustics*, 1986, 18, (3), pp. 137-145
- FINLEY, I.P., and WOOD, J.W.: 'An investigation of synthetic aperture radar autofocus'. RSRE memorandum 3790, April 1995, Ministry of Defense, RSRE Malvern, UK
- BLAKE, A.P.: 'Autofocus techniques: multilook registration and contrast optimisation - a comparison'. RSRE memorandum 4626, September 1995, Ministry of Defense, RSRE Malvern, UK.
- JOHNSON, K.A., HAYES, M.P., and GOUGH, P.T.: 'A method for estimating the subwavelength sway of a sonar tow-fish', *IEEE J. Ocean. Eng.*, Oct. 1995, 20, (4), pp. 258-267

VERIFICATION AND ANALYSIS

Verification

TOWARDS MULTI-CRITERIA DECISION SUPPORT FOR MODELLING AND SIMULATION

Eshan Rajabally
John Dalton

Engineering Design Centre
Armstrong Building
University of Newcastle
Newcastle upon Tyne NE1 7RU
United Kingdom
t.e.rajabally@ncl.ac.uk
john.dalton@ncl.ac.uk

Pratyush Sen

School of Marine Science and
Technology
Armstrong Building
University of Newcastle
Newcastle upon Tyne NE1 7RU
United Kingdom
pratyush.sen@ncl.ac.uk

Steve Whittle
Systems Process Research and
Development
BAE SYSTEMS Warton
Aerodrome
Preston PR4 1AX
United Kingdom
steve.whittle@baesystems.com

KEYWORDS

Accreditation, Credibility, Subject Matter Experts.

ABSTRACT

As our understanding of the world improves and as more complex products and processes are developed, effective modelling and simulation becomes increasingly difficult. Hence the need arises for decision support for modelling, which considers the numerous criteria required for successful, modern-day modelling and simulation. Important criteria are presented in this paper along with existing means to assist in dealing with an important few in isolation. It is proposed that these means be unified to provide a foundation for multi-criteria decision support for modelling and simulation. The types of disparate evidence that must be evaluated to provide such support are briefly reviewed and the role of modellers' judgements is highlighted. Hence means of judgement consensus are discussed along with biases to be expected in decision making. Addressing the issues presented in this paper contributes to the development of decision support for modelling and simulation that is being pursued by the authors.

INTRODUCTION

Modelling and simulation are becoming increasingly difficult to conduct appropriately, as our knowledge of the world expands. Furthermore, products and processes that we wish to model and simulate are becoming more complex. Models and simulations are subject to a range of uncertainties, variabilities and errors (Rajabally et al. 2002a) that make validation, verification and accreditation, onerous tasks. Indeed, such tasks have been suggested as deserving of computer support (Balci 1998; Lewis 1998). Validation

has been simply described as establishing whether the "right" model is being built with respect to modelling objectives whilst verification is establishing if the model is being built "right" (Balci 1998). Model accreditation is the official certification that a model is acceptable for use for a specific purpose (United States Department of Defense 1996).

Given the concerns mentioned, this paper identifies and addresses issues that contribute towards decision support for modelling and simulation. The wide range of criteria that must be considered are presented in the next section as an aid to modellers. The section after this surveys existing approaches to assist in dealing with certain criteria, which may be used to advance effective modelling. Then, in the section after this, important issues are presented that concern the consideration of disparate evidence in decision support for modelling and simulation. Finally, concluding remarks are given in the last section.

MULTI-CRITERIA IN MODELLING

In performing model accreditation, one must consider a wide range of criteria. Due to an endless variety of possible models and their purposes, accreditation is often an ad hoc procedure. Thus, the need arises for a decision support system to enable more rigorous and objective, selection and use of models. All the criteria considered for model certification contribute to the suitability of a model or its fitness for purpose. Model suitability or fitness for purpose have also been referred to as "total quality" (Balci 1998) or "acceptability" (Balci 2001) and provide model "candidate criteria" (Kilikauskas 2000). A list of important criteria is given below:

- accessibility,

- availability of model, input data and qualified operators/analysts,
- communicativeness,
- computational requirements,
- consistency,
- cost,
- deployability,
- device-independence/portability,
- execution time,
- extensibility/adaptability,
- fidelity,
- integrity/robustness,
- interoperability,
- legibility,
- maintainability,
- maturity,
- security considerations,
- self-containedness,
- self-descriptiveness,
- structuredness,
- testability,
- traceability, and
- usability/reusability.

The list is by no means exhaustive. Furthermore, some of the criteria may be considered to be interdependent and each criterion may have associated sub-criteria. For instance, Muessig (2000) defines “usability” as the “risk of simulation misuse” and proposes this to be related to the following:

- training in model use,
- training in results interpretation,
- documentation,
- technical support,
- user groups, and
- configuration management.

Meanwhile, Balci (2001) suggests that the total financial cost of modelling and simulation is dictated by the cost of:

- training,
- use,
- maintenance, and
- technical support.

It is generally a difficult task to satisfy all modelling criteria. For example, it may be expected that the cost of modelling and simulation would rise as usability increases. A bespoke computer model for an employee’s individual use would likely require significant investment in order to be formally deployed within their organisation. One must not only consider the financial costs listed of such a deployment, but also costs incurred in suitably developing the computer model to, say, satisfy the employer’s software standards. A compromise between the often-competing criteria of cost and usability is frequently required.

EXISTING MODELLING SUPPORT

Given the numerous criteria in modern modelling and simulation, efforts have been made to assist in model selection and use. The following three subsections present three important areas in which progress has been made. The

authors of this paper propose that aspects from these areas be unified in order to provide the foundation for progress towards multi-criteria decision support for modelling and simulation. Indeed, this is the focus of our work.

Credibility Assessment

Model credibility is a measure of confidence in the correctness or fidelity of a model and its appropriateness to the application of interest (Köster 2002). Several methodologies have been developed to aid modellers in demonstrating model credibility. For instance, Fossett et al. (1991) propose a framework of qualitative factors for assessing the credibility of military models and apply this to three competing models. In addition, Gass (1993) presents a numerical rating system for model accreditation that has been implemented in the software, EXPERT CHOICE. The system employs the Analytical Hierarchy Process (AHP) for rating, which is a ranking procedure based upon pair-wise comparisons. More recently, Balci (2001) describes the “Evaluation EnvironmentTM” software tool, which has been employed since 1999 in the United States National Missile Defence Program for model certification and also uses the AHP. Also, Brade and Köster (2001) propose “Credibility Indicator Trees” for assessing model credibility. Here, linguistic assessments of various model aspects are combined using multi-valued logic.

The authors of this paper have proposed a methodology for reasoning about user confidence in models that utilises Bayesian Belief Nets (BBNs), (Rajabally et al. 2002a and 2002b). A BBN is a directed acyclic graph that can be used for reasoning under uncertainty. Sensitivity analysis for BBNs can be performed automatically in some existing software tools, which enables the identification of the most significant sources of model concern. While this may be intuitive when dealing with individual models, it is less so if dealing with complex networks of interoperating models. The methodology thus allows resources that are used to improve complex models to be targeted more cost-effectively. The methodology has been implemented as a prototype software tool to support model use by BAE SYSTEMS in developing increasingly complex and integrated, military aircraft systems.

A Risk-Benefit Approach

The process of model accreditation is highly dependent on the nature of the purpose for which the model will be used. For example, a model of structural stress in a load-bearing vessel might be treated differently if the modelled vessel contains a toxic fluid rather than drinking water. For the two applications, the harmful consequences of the vessel bursting and its contents spilling, are vastly different if the vessel is located where people are present. Hence, the accreditation process is likely to be more stringent for the toxic fluid application rather than the drinking water application. Hence during accreditation one must explicitly consider the implications or consequences of using a model to assist in making practical decisions in the real world.

To assist in dealing with the consequences of using a model to make decisions, Muessig et al. (1997) propose a risk-benefit approach for determining required model credibility. The approach involves determining the overall positive or negative impact of a decision based upon a ranking of the magnitude and likelihood of the consequences in different domains. The impact is evaluated by combining the rankings using risk and benefit assessment matrices. For example, in the domain of “Personal Safety”, if a decision based on model use could result in death then the magnitude of negative impact is ranked as “Catastrophic”. If the potential occurrence of this negative impact could be ranked as “Frequent” then the risk in this particular application is judged to be “High”.

Muessig et al. (1997) promote their approach as a means of determining the appropriate mix of validation, verification and accreditation activities for a particular model purpose. Alternatively, the use of a model for a particular purpose that has been subject to a given mix of these activities may be justified or rejected using the approach. Hence it provides some objectivity for dealing with the model criterion of applicability/fitness for purpose/suitability and could be incorporated in decision support for modelling.

Fidelity Referents

The Fidelity Implementation Study Group of the United States Simulation Interoperability Standards Organization (SISO) defines fidelity as “the degree to which a model or simulation reproduces the state and behavior of a real world object or the perception of a real world object, feature, condition, or chosen standard in a measurable or perceivable manner; a measure of the realism of a model or simulation; faithfulness.” (Gross 1999).

It is due to our limited understanding of the real world that fidelity must be measured against the aspects of the real world that are indeed measurable. To achieve this, Roza et al. (2001) propose the use of a fidelity referent that they define as “a codified, structured, and formal specification of knowledge about what is commonly perceived, understood and accepted by a defined group of people to be reality, capable of serving as the comparative standard for reality correspondence assessment and associated activities of model or simulation development”. Furthermore, advice is given on the constituents of such a referent as well as guidelines to its formulation. The guidelines are illustrated by application to prototype air-traffic control and management models. The coordination of referent constituents should be considered in decision support for modelling and these constituents are summarised as:

- identity and management details (e.g. version number, revision history),
- applicability and status (e.g. domain specification, real-world coverage description),
- developer and validation agent information (e.g. expertise, qualifications of individuals),
- knowledge and data sources utilised (e.g. reference and contents description),

- real-world structural data (e.g. topology, characteristics, interactions of real-world objects),
- real-world behavioural and parametric data (e.g. initial conditions, dimensions and ranges of properties), and
- referent utility data (e.g. notation, terminology definitions).

CONSIDERING DISPARATE EVIDENCE

In striving towards multi-criteria decision support for modelling and simulation, one must inevitably consider disparate evidence. Means that allow this to some extent are considered in the previous section. However, they are highly specific and some general comments are now given in the following subsections.

Quantitative and Qualitative Information

There is a variety of “direct” quantitative indicators, regarding modelling and simulation, associated with the criteria given earlier. Examples concerning model fidelity are:

- convergence settings in an iterative model,
- dimensions of the computational grid or mesh in Finite Element Analysis and Computational Fluid Dynamics,
- numerical solution interpolation intervals,
- spatial step lengths in a discretised model,
- statistical measures of input data variability, and
- time periods in a time-varying model.

“Indirect” quantitative indicators may also be developed to assist in describing criteria in modelling and simulation. For instance, again with regard to model fidelity, Pace (1998) proposes the following indicators:

- number of “entities” modelled expressed as a fraction of those possible,
- depth with which “entities” are represented expressed as a fraction of the number of possible “levels” (e.g. system of systems, system, sub-system, component),
- “influences” upon “entities” expressed as a fraction of those possible, and
- “relationships” between “entities” expressed as a fraction of those possible.

Frequently, when conducting model accreditation, one relies upon qualitative information and subjective decisions. For instance, it is often accepted that validation against field test data is superior to validation against expert opinion (i.e. face validation), which in turn is superior to validation against other simulations (i.e. benchmarking), (Muessig 2000). Indeed, certifying that a model is appropriate for a particular purpose uses informal, ad-hoc reasoning by modellers based upon their experience and expertise (Rajabally 2002b). To consider the opinions of a number of modellers in such matters, there are means to arrive at an acceptable consensus and these are briefly surveyed in the next Subsection.

Expert Judgement Consensus

Approaches used for the aggregation or combination of multiple expert judgements to reach a consensus of opinion fall into two classes (Meyer and Booker 1991). The first class is loosely termed behavioural approaches in which the experts interact during an elicitation session using persuasion and compromise to reach a consensus. In the Delphi approach for example, successive iterations are used to reach one consensus. The second class of expert judgement aggregation uses mathematical approaches. In these approaches, functional combination rules are employed to return a consensus. Behavioural approaches suffer from problems of personality and group dynamics, whilst mathematical approaches may arrive at solutions that are not universally satisfactory (Ng and Abramson 1992). In our research, the authors of this paper have focussed on mathematical approaches because these are more amenable to development within a software tool and are generally less time consuming to use.

A range of mathematical approaches for expert judgement consensus exist. For those judgements that can be expressed as single-value probabilities, Ng and Abramson (1992) propose the use of linear and logarithmic opinion pools for generating a consensus. A linear opinion pool is a weighted sum of all expert probabilities while a logarithmic opinion pool is a weighted product of all expert probabilities. For imprecise judgements that can be captured by ranges of probabilities, Dempster-Shafer theory (Shafer 1976) may be used to arrive at a consensus. A range may be defined by “belief” and “plausibility”, which represent lower and upper bounds on probability respectively. Also, Herrera and Herrera-Verdina (1997) describe methods based on fuzzy set theory for the aggregation of weighted linguistic information. These methods can be used to combine expert judgements that are even more imprecise in nature. However, as Strigini (1996) notes, a consensus of expert judgements is never a guarantee of correctness.

Human Biases

Human judgement is subject to a range of biases and these should be considered to enable an objective and justified certification process for models. Meyer and Booker (1991) distinguish between two types of biases that arise during expert judgement making. The first is motivational bias, which occurs when the judgement elicitation process alters the experts’ reports of their opinions. For example, if an expert feels he or she has been led by an interviewer during face-to-face elicitation. The second is cognitive bias, which occurs when experts’ judgements fail to follow normative, statistical or logical rules (i.e. estimates based on reasoning that is not mathematically correct). In literature, motivational biases have received much less coverage than cognitive biases due to the interest sparked in the latter that one might attribute to the seminal research of Kahneman et al. (1982). Keeney and von Winterfeldt (1991) suggest three important cognitive biases to be overconfidence, anchoring and availability. Overconfidence is the tendency to express more certainty in judgements than is appropriate. Anchoring is the tendency to focus on an initial value for the elicitation and subsequent insufficient adjustment. Availability is the

tendency to overemphasise issues that are easily imagined or recalled. Two other relevant cognitive biases proposed by Meyer and Booker (1991) are impression management and wishful thinking. Impression management is termed a social pressure, which leads to experts attempting to respond in such a way that would be acceptable to their employers or colleagues. Wishful thinking is a bias also known as conflict of interest, which occurs when individuals’ hopes influence their judgement.

Subject matter experts are commonly employed for evaluating models and simulations for certification (Balci 2001). One can seek to reduce the presence of human bias by the careful selection of these experts. Meyer and Booker (1991) suggest selecting the opinions of experts who have the least to gain from their response. However, they continue that often those most qualified will be those with the most at stake. Possible solutions to this problem include obtaining opinions from experts with divergent views or from experts representing different organisations such as academia and industry. Pace (1999) reviews desirable attributes of subject matter experts, which include independence, competence, trustworthiness, having sound judgement and being objective focussed. It is also desirable that experts conform to relevant accreditation standards. Strigini (1996) proposes that experts may be assessed for the necessary conditions of good judgement such as consistency of opinion when repeatedly presented with the same evidence.

When expert judgement is required concerning model certification, one should seek to reduce the effects of human bias. Strigini (1996) suggests the following general means to help solve the problem of expert fallibility. One may attempt to:

- change the experts’ tasks to make them less error-prone,
- change the experts’ tasks to make them more amenable to analysis,
- help the experts in detailing their evidence, deductions and inferences,
- help the experts in finding and correcting fallacies in their reasoning, and
- make the best use of the availability of multiple experts.

Cognitive biases may be challenged by the decomposition of a given argument into individual inference steps and the consideration of these in isolation (Strigini 1996). Alternatively, one may ask experts for reasons why they might be wrong. Also, in order to verify expert judgements, “reasonableness checks” might be performed such as checking for the consistency in the opinions of different experts.

CONCLUSIONS

A range of issues concerning the tasks of model selection and use have been addressed in this paper. These tasks are complicated due to:

- the numerous criteria one must satisfy,
- the disparate sources of information one must consider, and
- the effects of human fallibility.

Despite existing research aimed at assisting modellers in these tasks, there remains no universally acceptable means of model accreditation. Indeed, accreditation remains a largely ad-hoc endeavour.

The authors of this paper have developed a methodology for reasoning about confidence in model use (Rajabally et al. 2002a; 2002b) that has been implemented as a prototype software tool. The methodology is being applied to case studies within BAE SYSTEMS to advance the effective use of modelling and simulation in military aircraft systems development. Our work is directed towards multi-criteria decision support for modelling and simulation, which addresses those issues reported in this paper.

REFERENCES

- Balci, O. 1998. "Verification, Validation, and Accreditation". In *Proceedings of the 1998 Winter Simulation Conference* (San Diego). 41-48.
- Balci, O. 2001. "A Methodology for Certification of Modeling and Simulation Applications". *Association for Computer Machinery Transactions on Modeling and Computer Simulation*, Vol. 11, No. 4, 352-377.
- Brade D. and A. Köster. 2001. "Risk-based Validation and Verification Levels Definition". In *Proceedings of the 1st European Simulation Interoperability Workshop* (London). 01E-SIW-013.
- Fossett, C.A., D. Harrison, H. Weintrob and S.I. Gass. 1991. "An Assessment Procedure for Simulation Models: A Case Study". *Operations Research*, Vol. 39, No. 5, 710-772.
- Gass, S.I. 1993. "Model accreditation: A rationale and process for determining a numerical rating". *European Journal of Operational Research* 66, 250-258.
- Gross, D.C. 1999. "Report from the Fidelity Implementation Study Group". In *Proceedings of the 1999 Spring Simulation Interoperability Workshop* (Orlando). 99S-SIW-167.
- Herrera, F., and E. Herrera-Weidma. 1997. "Aggregation Operators for Linguistic Weighted Information". *IEEE Transactions on Systems, Man and Cybernetics – Part A: Systems and Humans*, Vol. 27, No. 5, 646-655.
- Kahneman, D., P. Slovic, and A. Tversky. 1982. *Judgment under uncertainty: heuristics and biases*. Cambridge University Press.
- Keeney, R.L. and D. von Winterfeldt. 1991. "Eliciting Probabilities from Experts in Complex Technical Problems". *Transactions on Engineering Management*, Vol. 38, No. 3, 191-201.
- Kilikauskas, M.L. 2000. "Establishing the Credibility of Rolling Airframe Missile Simulations for Use in Operational Testing: Lessons in Process from a Success Story in VV&A". Working Group 29 of the 68th Military Operations Research Society Symposium, US Air Force Academy.
- Köster A. 2002. "Determination of Credibility in V&V Activities". In *Proceedings of the 16th European Simulation Multiconference* (Darmstadt). 262-266.
- Lewis, R.O. 1998. "A Practical Management Model and Metrics for Verification, Validation, and Accreditation (VV&A) Programs". In *Proceedings of the 1998 Fall Simulation Interoperability Workshop* (Orlando). 98F-SIW-012.
- Meyer, M., and J. Booker. 1991. "Eliciting and Analyzing Expert Judgement. A Practical Guide". In *Knowledge Based Systems Vol. 5*. Academic Press.
- Muessig, P.R. 2000. "An Integrated Approach to Evaluating Simulation Credibility". In *Proceedings of the 2000 Summer Computer Simulation Conference* (Vancouver).
- Muessig, P.R., D.R. Laack, and J.W. Wroblewski. 1997. "Optimizing the Selection of VV&A Activities, A Risk / Benefit Approach". In *Proceedings of the 1997 Winter Simulation Conference* (Atlanta). 60-66.
- Ng, K.-C., and B. Abramson. 1992. "Consensus Diagnosis: A Simulation Study". *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 22, No. 5, 916-928.
- Pace, D.K. 1998. "Dimensions and Attributes of Simulation Fidelity". In *Proceedings of the 1998 Fall Simulation Interoperability Workshop* (Orlando). 98F-SIW-017.
- Pace, D.K. 1999. "Use of Subject Matter Experts (SMEs) in Simulation Evaluation". In *Proceedings of the 1999 Fall Simulation Interoperability Workshop* (Orlando). 99F-SIW-018.
- Rajabally, E., P. Sen, and S. Whittle. 2002a. "A Methodology for Model Dependability Assessment". In *Proceedings of the 2002 Engineering Design Conference* (London). 291-300.
- Rajabally, E., P. Sen, and S. Whittle. 2002b. "Assessing Uncertainty in Models of Complex Systems". In *Proceedings of the 16th European Simulation Multiconference* (Darmstadt). 323-327.
- Roza, M., J. Voogd, and H. Jense. 2001. "Defining, Specifying and Developing Fidelity Referents". In *Proceedings of the 2001 European Simulation Interoperability Workshop* (Middlesex). 01E-SIW-008.
- Shafer, G. 1976. *A Mathematical Theory of Evidence*. Princeton University Press.
- Strigini, L. 1996. "Engineering Judgement in Reliability and Safety and its Limits: What can we learn from Research in Psychology". Centre for Software Reliability Technical Report, City University, London.
- United States Department of Defense. 1996. *DoD Modeling and Simulation (M&S) Verification, Validation, Accreditation (VV&A)*. DoD Instruction 5000.61.

AUTHOR BIOGRAPHY

ESHAN RAJABALLY was born in Liverpool, United Kingdom and studied aeronautical engineering at the University of Bath. He graduated with a Masters degree in June 2000 and now works as a Research Associate at the Newcastle Engineering Design Centre. His email address is <t.e.rajabally@ncl.ac.uk>.

PRATYUSH SEN is Professor of Marine Design and Construction at the University of Newcastle. His research interests include design synthesis tools, particularly the use of multiple criteria decision-making techniques. His email address is <pratyush.sen@ncl.ac.uk>.

STEVE WHITTLE graduated from the University of Wales with a BSc in Applied Mathematics in 1978. He has spent his entire professional career with BAE SYSTEMS. Much of this time has been spent developing models, particularly concerning high frequency radar cross signature of military aircraft. His email address is <steve.whittle@baesystems.com>.

JOHN DALTON gained his first degree in Mechanical Engineering at the University of Northumbria and his Masters Degree in Engineering Research at the University of Newcastle. He is currently a Research Associate at the Newcastle Engineering Design Centre. His email address is <john.dalton@ncl.ac.uk>.

LOGIC AND HYBRID FORMAL VERIFICATION TECHNIQUES FOR PROCESS PLANT CONTROL: A CASE STUDY

Luca Ferrarini, Claudio Maffezzoni and Francesco Schiavo
Dipartimento di Elettronica ed Informazione
Politecnico di Milano
20133, Milano, Italy
E-mail: {ferrarin,maffezzo,schiavo}@elet.polimi.it

KEYWORDS

Process control, formal verification, hybrid automata, model checking.

ABSTRACT

In the paper, the problem of formal verification of logic control systems for process plants is investigated. Innovative tools such as Spin (a logic model checker) and CheckMate (a Matlab toolbox based on hybrid automata) have been exploited. To critically prove the effectiveness of the verification techniques a case study drawn from the field of conventional power plant control, namely the condensate water extraction system, is considered. A methodology to represent the relevant control system models within the model checking framework has also been proposed.

INTRODUCTION

Advances in automation technology have led to the growing use of more and more sophisticated control systems for process plants, especially in power generation systems, oil and gas production platforms, nuclear and chemical reactors, boilers and furnaces [Park 1997]. The availability of information and communication technologies has permitted the fulfilment of complex requirements on plant safeness and management, but has also increased the overall system complexity. In particular, logic control systems for process plants have become very complex systems themselves, and often require very high dependability [De Smet et al. 2001]. Clearly, such a complexity growth must be supported by adequate validation techniques, besides the wide-spread “trial and error”, in order to systematically find and remove design errors in a more efficient way and in a shorter time [Bowen and Hinchey 1997].

On the contrary, the typical validation system applied to logic control systems for process plant control requires the development of the entire control system to connect it, through a bus, to a special command board, customly built or simulated via a virtual panel on a PC. The board is equipped with switches and push-buttons an operator can use to “manually” simulate the process under control, closing or opening the control loops and forcing values for some signals on the communication bus.

Such a technique has three major drawbacks:

- the choice of the “scenarios” to test is, to a relevant amount, up to the operator;

- there is no guarantee that the system is error-free;
- there are no means to understand where the errors causes lie (specification, design, implementation).

Alternative validation techniques are based on the simulation of the logic control system and of the plant for some scenarios [Carpanzano et al. 2002]. Although more flexible and useful in discovering many errors, simulation can not guarantee that the logic control system is error free. Formal verification could be a successful alternative approach, since it aims at mathematically proving that the designed system has all the specified functionalities and that no erroneous nor dangerous behaviour is possible. Although significant applications are still few [Havelund et al. 2001, Park 1997], the practice of such techniques is growing wider in industry [Heitmeyer 1998]. In particular, among the various formal verification techniques applicable to logic control systems, the ones based on model checking appear to be the most mature and the most effective [Clarke et al. 2000].

However, “classical” verification techniques address only logic variables and behaviours. Recently, new verification techniques have been developed for hybrid systems. They are drawing the attention of many researchers, since they will allow to integrate continuous (plant) and discrete (control) behaviours, with many expectations for the development of verification tools both effective and multi-domain [Silva et al. 2001].

The present paper addresses the problem of the verification of process control systems based on model checking techniques and on the use of hybrid systems. A specific case study drawn from the field of process plant control is considered, to test the real effectiveness of techniques and tools in industrial settings. Two commercial packages have been adopted to perform verification. Much attention is paid to understand what can be really gained by the control engineer and to the possibility to apply and generalise the verification methodology, i.e. to the hypotheses to construct the relevant models.

This paper is organized as follows: the case study is presented in the Section 2, while Section 3 proposes a possible formalization of the logic control verification problem as a model checking problem. The Sections 4 and 5 are dedicated, respectively, at defining procedures to model and verify logic control system within the Spin framework. In Section 6, the application of innovative hybrid system modelling and verification techniques are discussed with reference to the tool CheckMate. The major

results achieved and future research directions are collected in the last section.

THE CASE STUDY: WATER EXTRACTION SYSTEM

The case study for the application of formal verification techniques has been chosen to be an industry sized one. The study has been focused on the logic control schemes for the water extraction system in a conventional power plant. A schematic layout of the water extraction system is depicted in figure 1.

The main components of the system are a tank, two extraction branches and a recirculation line.

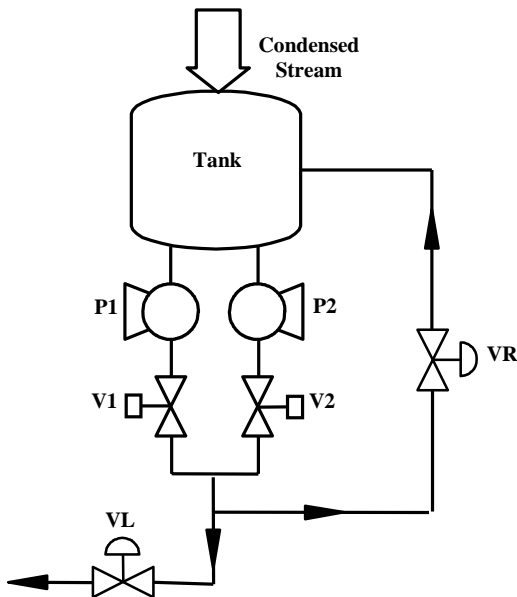


Figure 1: Scheme of the Water Extraction System

The input condensate stream is stored in the tank, where the level is held constant by a control system acting on the valve VL.

Two extraction branches depart from the tank; each branch is equipped with a manual on-off valve, a mechanical filter (both not represented, for simplicity), a pump (P1, P2) and a motorized valve (V1, V2).

A second control system prevents pump damages by acting on the valve on the recirculation line (VR), thus guaranteeing that the fluid flowing through the pumps is held above a minimum level.

Beside the continuous control systems illustrated above (level and minimum flow control), the plant is equipped with a complex logic control system in charge of operating the pumps P1/P2 and the valves V1/V2, which are used mainly for the extraction system start-up, shut-down or for emergency manoeuvres.

The logic control architecture is organized onto three hierarchical and modular levels (see figure 2):

- High level: control of the start-up and shut-down sequences for the entire system.
- Middle level: control of the start-up and shut-down sequences for each branch; coordination of

the signals to be transferred to the underlying pumps and valves.

- Low level (“drive” level): control of the physical devices (pumps and motorized valves).

Each logic module can exchange signals with the field, local and remote consoles, local pushbuttons or other logic modules (for the sake of clarity in figure 2 only the main signals exchanged by each module are reported).

The logic control system to be verified was available only in the form of control design schemes, made with classical function block formalism. Such schemes involved approximately 2000 components (logic gates, flip flops and timers) variously connected, about 350 relevant signals (inputs, outputs or internal communications) and 100 time varying parameters.

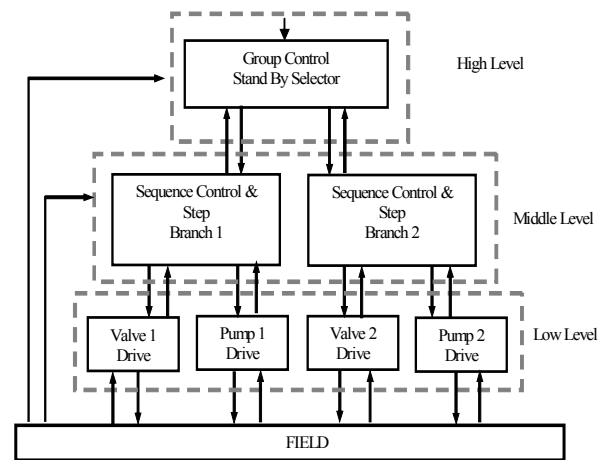


Figure 2: Logic Control Architecture

THE CONTROL SYSTEM VERIFICATION AS A MODEL CHECKING PROBLEM

The design schemes are a formal model of the system not directly analyzable. To formalize and formulate properly the verification problem is necessary to translate such a model into an easily verifiable “equivalent” one [Schiavo 2002].

Most of the models used for the formal verification of logic-based systems are based on *Kripke structures* [Clarke et al. 2000]. A Kripke structure is a mathematical structure, similar to an automaton, consisting in a set of states, a set of transitions between states and a function that labels each state with a set of logic attributes (called *properties*) which are true in that state. Paths in a Kripke structure (i.e. sequences of states) can represent the temporal evolutions of a system, while the information about the system conditions while evolving are carried by the state attributes. The initial state in the structure is determined on the basis of the initial conditions for the original system.

The temporal evolutions of a Kripke structure can be represented with a *computation tree*, which is formed by “unwinding” the structure into an infinite tree with the initial state at the root. The computation tree of a structure shows all the possible executions starting from the initial state. A computation tree is a suitable support to express properties to be verified for the corresponding Kripke structure and thus for the represented logic system [Pnueli 1977].

The properties to be verified can be expressed using a temporal logic such as CTL* (Computation Tree Logic) [Clarke et al. 2000]. CTL* formulas are composed of basic logic operators (*AND*, *OR*, *NOT*) and special operators: path quantifiers and temporal operators.

There exist two path quantifiers: **A** (“for all computation paths”) and **E** (“for some computation paths”). These operators can be used with reference to a particular state, in order to specify that all of the paths or some of the paths starting from that state must fulfil a specified property.

There are five basic temporal operators, but the most important ones for logic control verification are **F** (“eventually” or “in the future”) and **G** (“always”). These operators can be used with reference to a particular path to specify that a specific property must hold in at least one state or in all the states of such a path.

The problem of verifying the validity of a temporal formula over a Kripke structure is known as *model checking* problem. Such a problem is widely studied in the scientific community [Clarke et al. 2000].

The proposed process for formal verification of logic control systems comprises, thus, three phases: transforming the system model from design schemes to Kripke structures, expressing correctness requirements with temporal logic formulas and performing the verification using model checking techniques.

Once the logic control schemes have been “translated” into a model based on Kripke structures, a wide range of tools and algorithm for solving the model checking problem are available [Clarke et al. 2000]. However, a manual translation could be suitable only for small portions of the control schemes, highlighting the necessity for a technique to automate (or partially automate) such a process.

The following sections will then focus on the definition of an automatizable procedure to represent the control

schemes with a special tool (Spin) which exploits a special programming language (Promela). Promela programs can be automatically translated in equivalent models based on Kripke structures.

Within the model checking problem formulation depicted above, it is interesting to note that, despite the fact that the model and the temporal formulas are able to describe the temporal evolution of the system, the “time variable” is not directly involved: only the order of the events occurrence is relevant, but not the absolute time when they happened. It will be later showed that the lack of direct handling of time related variables is a major drawback of “classic” model checking techniques, thus denoting a possible future research direction, at least as far as model checking techniques for control system verification is concerned.

SYSTEM MODELLING WITH PROMELA/SPIN

The tool chosen for logic control verification is Spin [Holzmann 1997]. Spin is a freely distributed and widely used model checker, designed and implemented at the Bell Labs. Within Spin, the system to be verified is described by a program, using a language called Promela (see figure 3); the correctness requirements (i.e. the properties to be verified) are expressed with LTL formulae. LTL (Linear Temporal Logic) is a subset of CTL*, the main difference being that the operator **A** must precede every formula.

The modelling language Promela is syntactically similar to C, and it includes important features as support for nondeterministic execution, process concurrency and interprocess communication and synchronization. The Promela program (i.e. the system model) is automatically transformed into a structure that can be verified using model checking techniques.

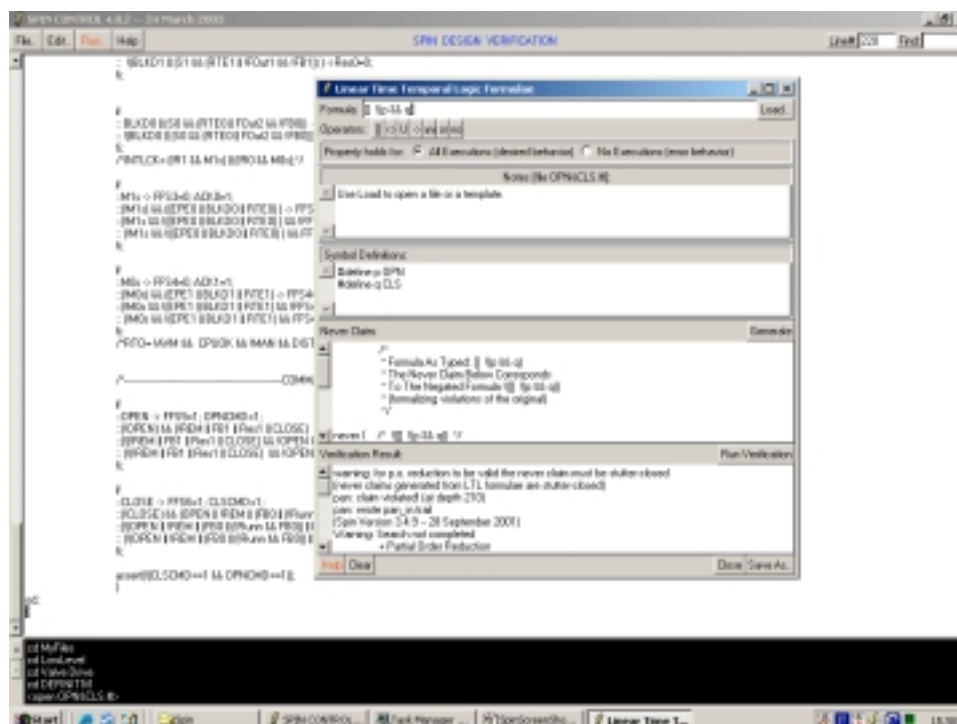


Figure 3: System Modelling with Spin

Spin includes simulation facilities; such a feature is useful to check the correctness for the principal system behaviours before performing any verification.

The verification process is completely automated and can produce either a positive result (i.e. the desired property is valid for the system) or a complete execution trail leading to the error configuration. Such a trail can be directly used for simulation, in order to let the user analyze the event sequence that led to the correctness requirement violation.

The verification algorithm uses the partial order reduction technique [Clarke et al. 2000]; such a technique is based on the consideration that very often the execution order of two (or more) concurrent instructions does not influence the validity of a specific property. It is thus possible to verify the validity of a specific property analyzing only classes of execution sequences, where each class represents an entire set (potentially with infinite cardinality) of execution sequences undistinguishable for the validity of that specific property.

Spin has been originally developed for the verification of communication protocols; nevertheless, its unique features soon extended its applications to various different fields, including logic control verification [Havelund et al. 2001].

The use of such an instrument in this field requires, however, particular attention in the modelling phase. In the present work a suitable modelling methodology has been investigated and applied. This methodology has been developed following the modularity and hierarchy paradigm typical of logic control systems, identifying, also on the basis of the design schemes, three different aggregation levels to be modelled:

- Modelling of basic logic components
- Modelling of logic control modules
- Modelling of intermodule communication

The modular modelling has led to the adoption of a corresponding “modular” verification approach (verification of single modules, of single levels and of the entire system) along with the identification of a set of properties to be verified at each stage.

Basic Logic Components Modelling

The logic components AND, OR, NOT and flip-flops can be easily modelled using standard constructs natively available within Promela. The behaviour of the timer component can be regarded as follows: the output signal assumes the value one if the input signal has been one (zero) continuously over a time interval, and zero otherwise. The modelling of timer components, however, revealed to be a critical issue, because Promela does not permit to explicitly model time.

The solution adopted is based on precise assumptions about the functioning of the components. As a matter of fact, each component is supposed to execute an infinite cycle with three phases:

- Reading of the inputs
- Computation
- Writing of the outputs

This model has the advantage to be both simple and close to the real functioning.

A timer component can then be easily modelled assuming a “conventional” time value for the execution of every cycle, so that is possible to measure time by tracking the number of executed cycles.

Logic Control Modules Modelling

The functioning model for the logic control modules is a three-phases cyclic one similar to the one shown above: inputs reading, computation, outputs writing.

The computation and outputs writing phases are obtained by composing within a single Promela program the models of the basic logic components, following the function block topology.

To complete the model, a mechanism is necessary to implement the inputs reading phase. As a matter of fact, at the beginning of each cycle a module should read logic signals coming from other modules, from operator interface or from the field itself.

The implementation of the inputs generation mechanism will be discussed separately, because of its complexity and of its strong connection with the verification strategy itself.

Intermodule Communication

The model for the communications between modules is based on the modules hierarchy adopted in the design schemes. That means that the modules at the higher level are executed first: they process their inputs and produce outputs that are fed to the lower levels modules which then start executing. Modules belonging to the same level are concurrently executed and their output generation phase is synchronized with the inputs reading phase of the lower modules.

Input Signals Generation

To simulate or verify a single module, it is necessary to generate somehow its input signals. As a matter of fact the same consideration extends to the entire system, since to perform simulation or verification it is necessary to generate some input signals (the ones coming from operator desk, from local pushbuttons, etc...).

The strategy that lies behind such a input signals generation is a critical issue to achieve a formal verification both effective and non time-consuming. An exhaustive generation of all the input combinations and sequences would lead to an unaffordable computational burden, but, on the other side, the choice of which input combinations and sequences to analyze should not be left to the user, because there would be no guarantee that all the relevant cases are investigated.

Achieving both the goals (full investigation and affordable computational burden) can be made possible exploiting some particular features available within Spin. As a matter of fact, the methodology adopted is based on the use of some peculiar Promela constructs (nondeterministic execution support) and on the partial order reduction technique that is used by Spin.

A detailed description of the Promela language and of the partial order reduction algorithm adopted is beyond the scope of this paper; however, a simple explanation will be

given here in order to let the reader understand the basic ideas behind.

Within code portions of Promela program, it is possible to randomly choose and execute a single instruction among a set of mutually exclusive and executable ones. Using this feature at the beginning of each cycle, it is possible to “potentially” generate all the input combinations and sequences for a logic module simply by inserting the two possible values (true, false) for each input signal in a set of mutually exclusive and executable instructions.

For example, consider the following portion of pseudo-code that generates the input signals **IN1** and **IN2** for a logic module:

```
beginCycle
{
    ChooseOneInstruction
    {
        IN1:=TRUE;
        IN1:=FALSE;

        IN2:=TRUE;
        IN2:=FALSE;
    }
    ...
}
endCycle
```

All the four assignment instructions for the two inputs are executable at the beginning of each cycle, but only one will be effectively executed, that is, at each cycle one input signal randomly chosen can (but is not forced to) change.

As a matter of fact, to potentially generate all the inputs combinations and sequences, every single input should be permitted to change at each cycle. To achieve such a result would imply to create a set of two executable mutually exclusive instructions for each input signal instead of a single set for all the signals as showed above. On the other side, such a solution would significantly increase the computational burden for the verification (each set added can double the number of scenarios needed to be analyzed). Some authors assume that, in every condition, only an input signal per cycle can change [Park 97].

The solution here adopted is a little different: the input signals have been divided into groups and, at each cycle, only a signal per group has the possibility to change; the inputs grouping can be based on the signals provenience (for example, it is unlikely that two signals coming from the operator change exactly within the same cycle, while this assumption does not hold while referring to an operator signal and a field signal).

The choice of how many groups the inputs should be divided in has to be taken carefully, being a trade-off between the generality of the model and the verification computational burden. Using such a strategy to model the generation of inputs signals, one can be relieved from the necessity to “imagine” or define the scenarios to be analyzed or testes, leaving such a duty to the specialized algorithms of the verification system.

Controlled Process Module

The logic control system is strictly connected with the controlled plant: their functioning depends, in some way, on their mutual influence; it is thus necessary to investigate

what classes of models of the controlled plant are the most suitable ones for formal verification.

Basically, there exist two possibilities: open loop (OL) models (that is only the logic control is modelled) or closed loop (CL) models (that is logic control, plant and their mutual influence are modelled).

The first solution (OL) has the advantage of being easier (no plant model is required at all) and it is suitable to perform verification of “structural” properties that the logic control system should have independently from the controlled plant. Examples of such properties are the “output liveness” (all the output logic signals should have the possibility to be activated) or the “command coherence” (it should not be possible to contemporary give opposite commands to an actuator, an emergency command should override all other commands, etc...).

The second solution (CL) has the advantage to be more “complete”, allowing the verification of properties depending on the mutual influence of the plant and the logic control system. An example of such a property could be the activation of a signal from the plant that informs the logic control system of the variations in some physical quantity in response to some previous command (as the signal “tank_level_low” after the pump has been working for some time). The drawback with the CL option is the plant modelling itself: first, it is not trivial itself in the general case, and second the signals between plant and control system are not independent from each other, but depend on the process itself.

A less complex CL solution can be obtained with the implementation of logic models only for the process components which do exchange signals with the control system, thus neglecting any other process components and any other interaction between process components.

Within the Promela/Spin framework, the latter CL solution can be the only one adopted, due to the limitations both in the language Promela mathematical support and in the verification system (which is really effective only when dealing exclusively with binary variables).

In this work, thus, the OL solution has been adopted for logic control modelling and verification with Promela/Spin, with the exception of simple logic models (CL) for the controlled valves and pumps (no physical interaction has been modelled between pump and valve, for example).

MODEL VERIFICATION WITH PROMELA/SPIN

In this section the problem of verifying the modelled system will be described. First, proper categories of properties to be verified will be identified and formalized, then, a modular verification procedure will be proposed and applied to the case study.

Temporal Formulae

The use of temporal formulae to express the desired properties for a logic control system allows a wide range of analysis to be conducted on the system itself [Clarke et al. 2000]. Nevertheless, it is suitable to divide the properties to be verified into categories, in order to identify sets of structural properties that a logic control system must have independently of the particular logic functions

implemented, along with sets of function-dependent properties.

Within this approach, the first category of properties involves the activation of the outputs: the logic control schemes should not stuck any output signal to the value zero or one. To perform the correspondent verification it is convenient to prove that the following temporal formulae are false (supposing **OUT** is a generic output signal):

- $\mathbf{A\ G\ (OUT)}$
- $\mathbf{A\ G\ (NOT\ OUT)}$

The validity of the first formula implies **OUT** assumes always the value one, for every computation path. Dually, the validity of the second one implies **OUT** assumes always the value zero, for every computation path.

A second important category of properties involves the commands used to activate and deactivate a device (like a pump or a valve) or to start and stop a logic sequence. Such commands should be coherent, that is they should not assume the value one at the same time. The temporal formula to express this is

- $\mathbf{A\ G\ (NOT(ON\ AND\ OFF))}$

The impossibility of generating contradictory commands should come together with the impossibility to reach logic incoherent states. For example, consider the sequences for the start-up and shut-down of the two condensate extraction branches: these sequences are composed of several steps and it is supposed that, at every instant, only one step could be active. Indicating with the logic variable $\mathbf{A_i}$ the activation of the i -th step in the sequence (supposed of 3 steps), it is necessary to prove the validity of the following temporal formula:

- $\mathbf{A\ G\ (NOT(A_1\ AND\ A_2)\ AND\ NOT(A_1\ AND\ A_3)\ AND\ NOT(A_2\ AND\ A_3))}$

The properties identified so far are aimed at verifying the absence of erroneous/dangerous behaviours. However, it is important to introduce also properties aimed at verifying the presence of the desired behaviours in some specific situations. For example, it is desired that, whenever an valve emergency closing command is issued, the logic control executes it, whatever different command may come from operator interface or local push-buttons. The temporal formula to express this requirement is as follows:

- $\mathbf{A\ G\ (f \Rightarrow (F\ g))}$

In the formula, f represents the logic conditions that should cause the control to do something represented with the logic conditions g . The validity of the formula requires that, for every computational path, if exist a state in which f holds, then there exist also a state in which g holds.

Verification Procedure

The verification procedure should follow the modular paradigm used in the modelling (and derived from the design schemes), so it has been divided into three sequential phases:

- single module verification
- level verification
- overall logic control verification

Such a division could seem pointless, since the final aim is the verification of the entire logic control (i.e. only the third phase). Nevertheless, operating with phases of increasing complexity can help in tackling the difficulties of the entire

problem. The first two phases, as a matter of fact, can identify errors and inconsistencies for a relatively small analysis set, permitting to easily track back errors' trails, and allowing, in the third phase, to concentrate only on the interactions between the logic modules.

Verification Results

For the introduced case study, the complete model for the logic control system has been implemented, following the methodology illustrated in the previous sections. A complete set of properties to be verified has also been identified, including both "structural" and functional requirements.

The verification has led to the following results:

- Single logic modules verification
 - Big number of properties verified
 - Quick computation (< 15 sec)
- Level verification
 - Big number of properties verified
 - Significant grow of the time required (≈ 20 min)
- Overall system verification
 - Computational difficulties

The results for the first two phases have been really successful, with the identification and correction of numerous subtle bugs, while the computational complexity affected the results for the remaining verifications.

Despite the computational problems, deriving also from the complexity of the real system, the formal analysis performed permitted to effectively improve the logic control design, identifying many possible malfunctioning cases whose causes would have been really hard to track back for the installed system.

HYBRID MODELLING, SIMULATION AND VERIFICATION TECHNIQUES

The major drawbacks emerged using Spin are the impossibility of including a realistic plant model and the lack of a time metric, that is the impossibility of dealing explicitly with time and time dependant variables.

A possible solution to overcome those drawbacks could be the adoption of new tools applying some emerging techniques for hybrid systems modelling and verification [Bemporad and Morari 1999, Silva et al. 2001].

In this context, a hybrid system is intended to be a complex system whose behaviour is influenced by events and comprising logic (or discrete) components and component whose functioning is governed by a continuous dynamic.

Among the numerous emerging tools, *CheckMate* [Silva et al. 2000] is one of the most promising and has been chosen to try a different approach to the verification problem for the case study.

CheckMate Overview

CheckMate is a tool for modelling, prototyping, simulating and formally verifying hybrid dynamic systems; it is based on Matlab/Simulink and the Stateflow toolbox from The MathWorks, Inc.

CheckMate has been designed for a specific class of hybrid systems, called TEDHS (*Threshold Event Driven Hybrid*

Systems). In TEDHS the dynamic of the system can switch among a finite set of possible dynamics; the dynamic is switched according to a discrete signal produced by an automaton; the automaton behaves on the basis of the events generated when the state variables cross predefined thresholds.

An hybrid system model in CheckMate has continuous dynamics described by ordinary differential equations (possibly nonlinear), planar switching surfaces, and discrete dynamics modelled by a finite state machine. The verification is performed using finite-state approximations known in the literature as quotient transition systems. The approximations are conservative in the sense that they capture all possible behaviours of the hybrid system.

The verification result, if not positive, can be inconclusive: the user is given the option to refine the current system approximation and attempt the verification again. Actually there exists also the possibility of a definitive negative result, when the approximating quotient transition system is proved to be equivalent to the original system [Silva et al. 2000].

The hybrid system model is composed of three distinct part: the continuous process, the logic control and the event generator, which constitutes the interconnection layer.

Continuous Process Model

The controlled process has to be represented as a *Switched Continuous System* (SCS); a SCS is a system with continuous dynamics and (nonlinear) state equations which are selected according to the value of a discrete input signal (u). The system evolves with a dynamic of the type $\dot{x} = f_u(x, t)$, where the specific function $f_u(\cdot, \cdot)$ is chosen among a finite set on the base of the value of the discrete input u . This representation requires an explicit formulation for the equation of the plant components as pumps and valves.

The following simplifications have been taken into account:

- The plant fluid is water at a constant temperature.
- The plant has only one extraction branch.
- All the pipes, filters and heat exchangers can be regarded as lumped head losses

The implemented model is based on the physical models for the process devices (pumps, valves, tanks, head losses, etc...) taking into account also the action of the modulating control (i.e. the level control acting on the level valve and the recirculation control acting on the recirculation valve).

The implementation of an explicit model has been a critical issue, since the simple collection of the devices models would have resulted into an implicit global model.

All the values needed for the model set-up have been calculated trough a different, more complicated and detailed simulator of the plant.

Logic Control Model

The logic control system has to be represented as a *finite state machine* (FSM) using Stateflow, a Matlab toolbox. The logic control system thus modelled receives as inputs the events generated within the interconnection layer between the SCS and the FSM itself. These events are

based on the time evolution of the state variable. The FSM produces the discrete output signal (u) to be fed to the SCS. A complete representation of the logic control system as a FSM would have been an unworthy modelling effort within this case study, so it has been decided to represent only on a logic control portion where the interactions between the controlled process and the logic control were most relevant. The FSM implemented is then the model of the sequence control governing the extraction branch start-up.

The sequence is composed of four steps:

- Valve Closing
- Pump Starting
- Valve Opening
- Sequence Check

The first three steps (their self composed of complex logic schemes) are the most relevant for the interaction between the process and the logic control, while the forth one has only the function to check the sequence has terminated correctly and, in case, to raise a system alarm. The entire sequence and every single step are associated with a time interval within which they must terminate their actions ("watch-dog").

The FSM is thus composed of five macro-states: one for each step in the sequence and one to represent error conditions (such as timeouts).

Interconnection Layer

The SCS and the FSM exchange data and events: the FSM generates the discrete control signal u for the SCS, while the inputs events which govern the evolution of the FSM are generated on the basis of the evolution of the state variables (x).

The event generator (i.e. the interconnection layer) is composed of polyhedral thresholds: defining one or more polyhedral region of the type $Cx \leq d$ it is possible to generate events associated with the state variable entering or leaving such regions.

Four events have been defined for the case study:

- *clk_out* (single step or entire sequence timeout)
- *valve_closed* (the valve has been closed)
- *valve_opened* (the valve has been opened)
- *pump_on* (the pump has been switched on)

Within CheckMate a fifth event is default available:

- *out_of_bound* (the state variables have left the defined analysis region)

Hybrid System Simulation and Verification

The verification for the modelled hybrid system has been preceded by some simulations. The simulations have been performed to gain confidence about the meaningfulness of some model parameters and behaviours with respect to the results obtained with other simulators much more complicated. In detail, it has been analyzed the temporal evolution of the fluid level in the tank with respect to the timing for the valve open/close operations and the pump start-up/shut-down. The results obtained were consistent. The verification were aimed at proving the following property:

- $(\mathbf{AG}(\sim fsm == err)) \& (\mathbf{AG}(\sim out_of_bound))$

The property holds true only if the error state (*err*) is unreachable and the state variables do not leave the analysis region (which represents only physical limitations).

The property proved true and the verification took about three minutes to finish.

It is important to notice that the verification has been performed on a model where both the modulating control (level control) and the logic control (sequence control) could operate at the same time.

Although quite promising, the experience with techniques for hybrid systems modelling and verification revealed some major drawbacks, along with great improvement possibilities.

The first negative aspect involves the modelling capabilities: to build an explicit model for the process it is very often necessary (as it was for the case study) to accept some important simplification. In addition, such a model is very often scarcely reusable, causing the need for a complete remodelling also for simple process changes.

A second negative aspect relies in the (conceptual and practical) difficulties of building a detailed model for the entire logic control as a single FSM as it is necessary using CheckMate.

A third criticisable aspect is the impossibility of modelling any operator intervention.

Last, but not least, it has to be noted that, whenever a property is proved to be false, no counterexample is given. Nevertheless, the drawbacks pointed out do not have to obscure the great innovations that CheckMate (and other tools for hybrid systems modelling and verification) brings: the explicit modelling of time, the fact that the controlled process, the modulating control and the logic control system are modelled within the same framework and, most important, the proposition of new verification techniques taking into account both the logic and the numerical aspects for the system model.

CONCLUDING REMARKS AND FUTURE WORK

In the paper, the application of formal verification techniques, and in particular of model checking, to logic control systems for process plants has been investigated and discussed. An industrial-size case study has been chosen, in order to check the real applicability and effectiveness of the proposed verification methodology. A specific tool, namely Spin, has been exploited for the verification, and specific guidelines have been proposed to build up all the relevant models. A set of categories of properties to verify has been also identified and proposed, to be used for process control systems. However, it is authors' opinion that the proposed verification methodology can be easily extended to other tools and applications.

The obtained results are encouraging from the theoretical point of view. However, some drawbacks, mainly due to the computational burden and to time variables modelling, limit the applications of such techniques in industrial settings.

The emerging techniques for hybrid systems modelling and verification appear to be more promising, at least for precise numerical analysis about the system itself.

Future work will aim at a further exploitation and formalization of the modular modelling and modular

verification technique with Spin, in order to allow the verification of some properties to single modules instead of the overall system, with clear advantages with respect to the computation burden. Future research will include also a study on object oriented extensions for hybrid systems modelling procedures.

REFERENCES

- Bemporad, A. and M. Morari, 1999. "Control of systems integrating logic, dynamics, and constraints", *Automatica* 35, no. 3, 407-427.
- Bowen, J.P. and M.G. Hinchey, 1997. "The use of industrial-strength formal methods", In *Proceedings of the Computer Software and Applications Conference* (Aug 11-15). IEEE.
- E. Carpanzano, L. Ferrarini, C. Maffezzoni, 2002. "A Bottom-Up Methodology For Testing Complex Control Functions Of Process And Power Plants", In *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain, 2492-2496.
- Clarke, E. M. and J. M. Wing, 1996 "Formal methods: state of the art and future directions", *ACM Computing Surveys (CSUR)*, Volume 28, Issue 4.
- Clarke, E. M.; O. Grumberg; and D. A. Peled, 2000. *Model Checking*. The MIT press, Cambridge, Massachusetts.
- De Smet, O., J.J. Lesage and J.M. Roussel, 2000. "Formal verification of industrial control systems". In *Proceedings of the 10th IFAC Symposium on Information Control Problems in Manufacturing* (Vienna University of Technology, Vienna, Austria), vol. 1, 125-132.
- Havelund, K., M. R. Lowry and J. Penix, 2001. "Formal analysis of a space-craft controller using SPIN", *Software Engineering* 27, no. 8, 1072-1095.
- Heitmeyer, C. L., 1998. "On the need for practical formal methods". In *Proceedings of the 5th Intern. Symposium on Formal Techniques in Real-Time and Real-Time Fault-Tolerant Systems* (Lyngby, Denmark,), 18-26.
- Holzmann, G. J, 1997. "The model checker SPIN", *Software Engineering* 23, no. 5, 279-295.
- Park, T., 1997. "Formal verification and dynamic validation of logic-based control systems", Ph.D. thesis, Massachusetts Institute of Technology, 1997.
- Pnueli, A. 1977. "The temporal logic of programs". In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society Press, 1977, 46-57.
- F. Schiavo, 2002. "Tecniche di verifica formale per le logiche di controllo di impianti di processo", Tesi di Laurea, Politecnico di Milano, a.a. 2001/2002 (in italian).
- Silva, B. I., K. Richeson, B. H. Krogh and A. Chutinan, 2000. "Modeling and verification of hybrid dynamical systems using CheckMate", ADPM 2000, September 2000.
- Silva, B. I., K. Richeson, B. H. Krogh and S. Engell, 2001. "An assessment of the current status of algorithmic approaches to the verification of hybrid systems". In *Proceedings of the 40th IEEE Conference on Decision and Control* (Orlando, Florida), 2001, pp 2867-2874, IEEE.

Analysis in Engineering

A CASE STUDY ON COMPARATIVE ECO-BALANCES FOR COATING MATERIALS WITH EPOXIDISED DERIVATIVES

Jorge Marx Gómez, Jana Görmer, Chaouki Khatib
Technical University of Clausthal
Institute of Informatics
Julius-Albert-Str. 4
38678 Clausthal-Zellerfeld
Germany
Tel: +49-5323-72-7115
Fax: +49-5323-72-7139
Email: gomez@in.tu-clausthal.de

KEYWORDS

Balance Valuation, Comparative Eco-balance, Life Cycle Assessment, Material Flow Networks

ABSTRACT

The scope of the presented project was to illuminate and describe the potential environmental impacts which are caused by the production and application of a coating material using epoxidised derivatives. Therefore an appropriate comparative eco-balance in collaboration with the company BIOMEL GmbH located in Dessau (Germany) shall be conducted. Problem definition, approach and results will be given.

INTRODUCTION

Objectives

In the field of coating materials volatile organic compounds (VOC) represents one of the highest environmental burden. Due to this fact the company BIOMEL GmbH started to develop a new product which should substitute VOC from coating materials. The new product has a pure native basis utilising the seed oil of the so called dragon-figurehead plant. The "reactive diluent" shall substitute VOC-containing solvents and because of its specific properties it can be used as a solvent as well as a binder.

In order to verify the environment-friendly behaviour of a coating material using the reactive diluent it is necessary to compare it with an other coating material concerning environmental impacts. It will be exemplified with the use of a concrete coating material containing solvent compared to a solvent-free concrete coating material (containing the reactive diluent). Therefore the method of eco-balancing has been chosen. Two eco-balances have been created and opposed (comparative eco-balance) according to ISO 14040 in order to obtain and derive meaningful and reliable environmental information.

Eco-balancing

Eco-balancing is a method for documenting the environmental aspects and potential environmental effects associated with a product or production process. This is done by:

- compiling an actual balance of relevant input and output flows of a production system,
- evaluating the potential environmental effects associated with these inputs and outputs,
- interpreting the results of the actual balance and effects with regard to the objective of the study.

As a result eco-balances serve and support the numerical registration and the interpretation of environmental effects. The expression „eco-balance“ is a generic term for a number of different methods for the numerical registration of the environmental effects (Seuring 1998), Rautenstrauch 1999). The following types of eco-balances can be named:

- Company Eco-Balance: This eco-balance is related to a place of production or business, i.e. to an organizational or operational unit. This eco-balance is regarded as a „Black Box“ since no analysis of the in-house operations takes place. The balance concentrates on the company's input and output of energy and materials and is frequently referred to as the basis of environmental reports.
- Process Balance: This eco-balance examines the flows of individual production steps specific to operation and makes a more precise look at the operational activities possible. Thus the degree of ecological harmfulness of the individual processes can be recognized in the total environmental pollution and gaps and potentials for optimization can be easily localized from this estimate.
- LCA: The product-related eco-balance (Life Cycle Assessment, LCA) balances the environmental effects of a product over its entire life, i.e. starting from the generation of energy/recovery of raw materials, preproduction, production, transport, distribution, product use or consumption and up to disposal.
- Substance Balance: Substance balances show lasting environmental use and environmental disturbances. Surface consumption, cultivation, and operationally-dependent landscape excavation belong to the group of enduring environmental use. Pollution of soil or ground water by the plant constitute enduring disturbances of

the environment. Furthermore capital assets and inventory present in the plant are evaluated according to ecological criteria.

Material Flow Networks

A particularly flexible and efficient approach to represent material and energy flows of a system is the application of material flow networks (Möller 1995). The systems can be production locations and operations of any size as well as stations in the life of the product. Consequently material flow networks are equally suitable for the preparation of company balances as well as product balances. This combination of company and product balances in one approach is one of the most important special features of the material flow networks. Material flow networks are suitable for modelling material and energy flows which originate between the system elements (e.g. between individual stages of an extensive production process) for a specific part of a real system. They thus create the basis for the preparation of eco-balances (Schmidt 1997). Material flow networks serve the preparation of eco-balances by listing balance-like the materials and the corresponding amounts, which are exchanged with the system environment through the inlets and outlets of the network. In material flow networks two types of procedures are principally taken into account and differentiated:

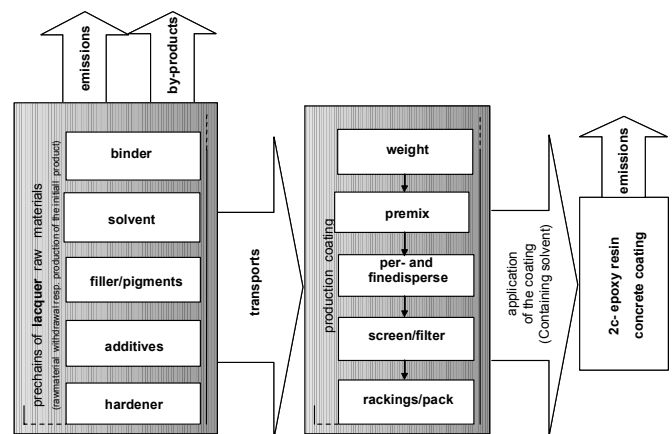
- Processes for the Transformation of Materials (transitions, in the network symbolized by rectangles): In these locations materials are converted with the help of energy. Every transition is on the one hand connected with a place from which they receive materials and on the other hand with places where they deliver materials. For every transition, how the inlet and outlet flows depend on one another can be correspondingly determined in the process shown. At the same time both the type of the materials affected and the mutual dependencies of the particular quantities can be specified.
- (Temporary) Storage of Materials (places, in the network symbolized by circles): Places are locations in the material flow networks where no conversion of materials occurs. They function as material storage for example; they are inlet and outlet points where material flows enter or leave the system. In view of the different functions the places can assume in the material flow networks, Umberto differentiates between the place types of input-places (inlet point), output-places (outlet point), storage-places (storage) and connection-places (direct throughput without storage).

Besides the aforementioned network elements further symbols are used in Umberto which also appear in the models here. In particular these are elements which are used by using sub-networks. Sub-networks are used in the network transformation for refinement as well as generalization in order to achieve a certain clarity in the network modelled. A hierarchical structure results through the creation of a network „behind“ an existing transition. If places in the overriding network are in conjunction with a transition, which contains a sub-network, these places appear

as network transition points in the sub-network. This applies to the places of the type Input, Output and Connection. The places and transitions are linked with one another by connections (links) so that a new network structure results. Connections lead either from a transition to a place or from a place to a transition. A connection between two similar network elements is not permitted. The tip of the pointer specifies the direction of the flows; a flow in the opposite direction is not permissible.

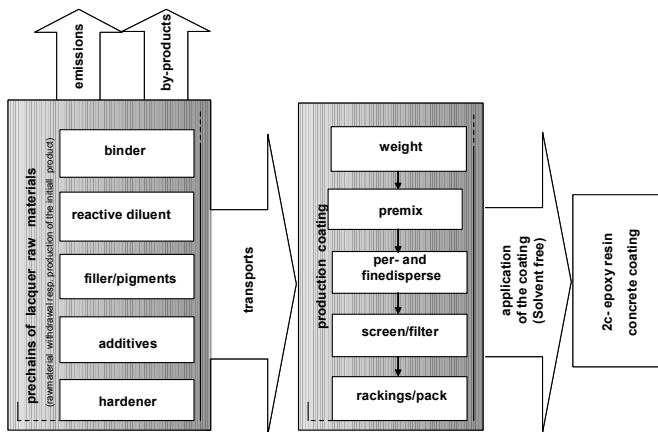
PROCESS STRUCTURE

In accordance to the investigation subject of an eco-balance, the life-cycle of the coating compound should be considered and investigated including all material and energy flows during the whole product life-cycle starting from the raw material extraction, along the production process, the utilisation and the disposal (Rautenstrauch 1999). Up to now only the first three phases have been considered. The disposal phase has not been covered due to identical removal treatments for both materials. Here the coatings will be removed and transferred to a hazardous waste disposal site. Furthermore it should be mentioned that no further relevant data are available for this. Both, concrete surface coatings (2K-epoxy-resin concrete coating, solvent-free and solvent-containing) consist of three layers: the grounding, the interface layer and the sealing coat. Each of them is manufactured in the same manner. The solvent-containing layers as illustrated in figure 1 are composed of the following precursors (Brock/Groteklaes/Mischke 1999): non-volatile fraction (binder, pigment and filling material, non-volatile additives), hardener and volatile fraction (solvent, dispersing agent, volatile additives).



Figures 1: 2c-epoxy resin concrete coating (containing solvent)

Layers of solvent-free concrete surface coating include binder, pigment and filling material, additives and hardener as well. Instead of the volatile solvent they include the reactive diluent (see figure 2).



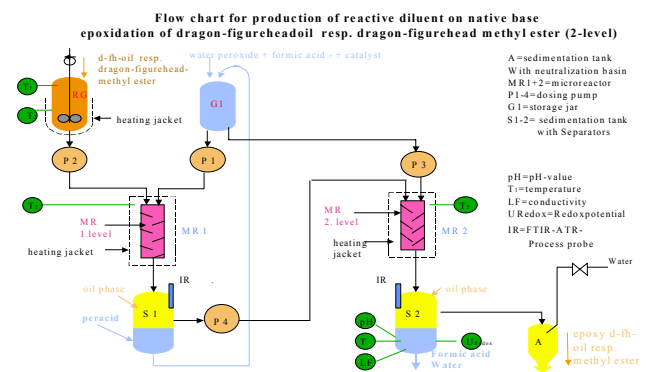
Figures 2: 2c-epoxy resin concrete coating (solvent free)

The production of both coating materials, i.e. the production of the respective layers will be conducted as follows: weighing out the raw materials for the dispersing approach, master batch (crude homogenisation), pre-dispersing, fine-dispersing (main-dispersing), filtrating, packaging and storage of the finished products (Brock/Groteklaes/Mischke 1999). Within the utilisation phase of the concrete surface coating the volatile part (solvent) of the solvent-containing concrete surface coating evaporates during the generation of the film without a chemical transformation (Baumann/Muth 1997). In contrast to this the reactive diluent of the solvent-free concrete surface coating is not evaporating. Due to a chemical reaction during the film generation the reactive diluent loses its diluent properties and becomes part of the binder.

DATA COLLECTION AND INTEGRATION

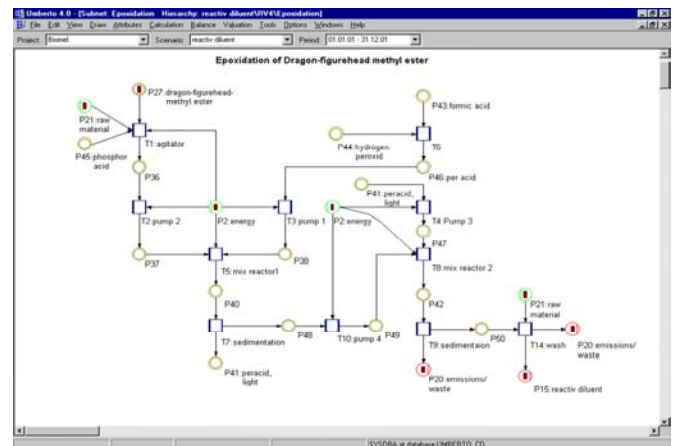
For the eco-balancing it is necessary to ascertain precise data about all involved processes. After the modelling of the process structure the overall model must be completed with the ascertained data in order to calculate the input and output flows and to generate the actual balance. The methodology of eco-balancing which provides the foundation for this study is based upon material flow networks (Schmidt,Häuslein 1997). Technical information about processes are modelled, specified and evaluated with the utilisation of the software tool Umberto.

E.g., modelling the sub-process “dragon-figurehead methyl-ester to epoxy dragon-figurehead methyl-ester” resulting in reactive diluent could be achieved using a detailed flow process chart with the corresponding data (see figure 3). Processes for the transformation of materials and energy are modelled in the network with transitions symbolised by rectangles. Storage and distribution of materials occur in places symbolised by circles.



Figures 3: Flow chart for a sub-process (epoxidation of dragon-figurehead methyl-ester)

In order to be able to specify places and transitions, the belonging material flows have been assigned to them. After having specified all relevant transitions and places, the material flow network could be calculated. Figure 4 shows exemplarily the epoxidation process modelled with the use of a material flow network.

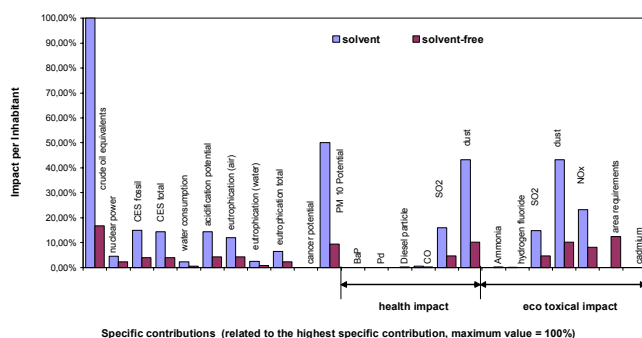


Figures 4: Umberto model of the epoxidation process

INVENTORY ANALYSIS, IMPACT ASSESSMENT AND BALANCE VALUATION

Based on the calculation mentioned above the impact assessment and balance valuation could be done using coefficient systems implemented in Umberto. The impact assessment is executed to consolidate the information contained in the actual balances for preparation of the subsequent balance valuations. In order to achieve this the documented material flows are to be described with regard to their potential environmental impacts. For substances with comparable impacts an aggregation is carried out. This procedure strives for a comprehensive impact assessment of all input and output flows. With the use of the software tool Umberto® it is possible to compare process balances immediately. Consequently significant differences between created balances (solvent-free and solvent-containing concrete surface coating) can be achieved. Impact assessment and balance valuation were conducted in several steps applying the UBA-method from the Federal Office for Environment (Umweltbundesamt 2000). In the first step of the impact balance, the classification, all material and energy

flows that have been confirmed in the actual balance are linked to impact categories according to their environmental impact potential, respectively. The impact potentials are calculated in the second step resulting in the characterisation. Modelling and aggregation of the actual balance data within the impact categories have been conducted. Within the categories suitable rates are determined which enable impact related aggregation. They clearly describe the potentials in a quantitative form (DIN 1998). Ten environmental impact categories have been suggested: consumption of raw materials, greenhouse effect, generation of photo-oxidants, acidification of soil and surface water, eutrophication, impediments to human health (cancer risk), direct damage to organisms and eco-systems (fine dust generation), introduction of nutrients into soil and surface water, consumption of space, noise incrimination (see figure 5).



Figures 5: Specific contributions of the impact categories of the coating system by comparison

In order to obtain a better comparability, the results have been normalised for inhabitant mean values in Germany (DIN 1999) and based on that, the relative specific contributions for impact categories have been calculated. The maximum specific contribution is provoked by the consumption of crude oil equivalents in the impact category “consumption of raw materials” and is set equivalent to 100%. All other specific contributions are to be set in relation to this value. Compared to the crude oil consumption caused by coating materials with conventional solvents, a specific contribution close to 17% in the category “consumption of raw materials” for the coating material based on the reactive diluent could be found. At the same time the consumption of crude oil equivalents gives the maximum specific contribution resulting from the solvent-free coating material. As far as the fine dust generating potential (PM10-potential) is concerned, the specific contribution could be reduced from 50% for the solvent-containing coating material down to 10% for the solvent-free coating material. The application of the impact categories “impediments to human health” and “direct damage to organisms and eco-systems” leads to a reduction of the specific contribution for dust from 43% (solvent-containing surface coating) to about 10% (solvent-free surface coating). Similar results could be obtained for the remaining impact categories. Merely, as far as the category “consumption of space” is concerned, better results could be obtained for the

solvent-containing coating material. The reason for this fact is the huge amount of dragon figurehead oil required in the reactive diluent production. This oil will be extracted from the seed of the dragon figurehead plant. The plant needs an appropriate agricultural cultivable area.

CONCLUSION AND OUTLOOK

Two comparative eco-balances for two production processes have been prepared in order to illuminate and describe the potential environmental impacts caused by the production. The first eco-balance has been conducted for coating materials whose solvents were build up on the basis of epoxy resin. The second eco-balance covers a coating material with an reactive diluent as solvent which is currently under development. The reactive diluent will be produced on a pure native basis utilising the seed oil of the so called dragon-figurehead plant. The obtained results of this case study clearly demonstrate that concrete coating materials on a native basis, i.e. without VOC-containing solvents are significantly more environment-friendly than the conventional solvent-containing concrete surface coating materials. Results were discussed with process engineers and industrial chemists. They acknowledged that this would correspond to their experiences and expectations.

REFERENCES

- Baumann, W. / Muth, A. (1997): Farben und Lacke: Daten und Fakten zum Umweltschutz mit 157 Tabellen. Springer, Berlin.
- Brock, T. / Groteklaes, M. / Mischke, P. (1998): Lehrbuch der Lacktechnologie. Vincentz, Hannover.
- DIN (1999): DIN 14042. Ökobilanz – Wirkungsabschätzung. Deutsche Fassung der prEN ISO 14042. Beuth Verlag, Berlin.
- DIN (1998): DIN 14043. Ökobilanz - Auswertung. Deutsche Fassung der prEN ISO 14043. Beuth Verlag, Berlin.
- Möller, A.; Rolf, A. (1995): Methodische Ansätze zur Erstellung von Stoffstromanalysen unter besonderer Berücksichtigung von Petri-Netzen. In: Schmidt, M.; Schorb, A. (1995), pp. 33-58.
- Rautenstrauch, C. (1999): Betriebliche Umweltinformationssysteme: Grundlagen, Konzepte und Systeme. Springer, Berlin.
- Schmidt, M. (1997): Stoffstromnetze zwischen produktbezogener und betrieblicher Ökobilanzierung. In: Schmidt, M.; Häuslein, A. (1997), pp. 11-24.
- Schmidt, M. / Häuslein, A. (1997): Ökobilanzierung mit Computerunterstützung: Produktbilanzen und betriebliche Bilanzen mit dem Programm Umberto®. Springer, Berlin.
- Seuring, S. (1998): Betriebliche Ökobilanzen. Grundlagen zu Begriffen, Methoden und Anwendungen. In: Sietz, M. (Ed.) (1998), pp. 65-114.
- Umweltbundesamt (2000): Hintergrundpapier „Handreichung Bewertung in Ökobilanzen“, Umweltbundesamt, August 2000.

BIOGRAPHY

Jorge Marx Gómez studied industrial engineering at the TFH Berlin. He is a lecturer and researcher at the Otto-von-Guericke-Universität Magdeburg where he obtained a PhD degree in business information systems with the work Computer-based Approach to Forecast Returns of Scrapped Products. Actually he is a visiting professor for business informatics at the Technical University of Clausthal.

ANALYSIS OF NEW PROPOSALS FOR PARALLEL REPLICATION BASED DISCRETE-EVENT SIMULATION

Àngel Perles, Xavier Molero, Antonio Martí, Juan José Serrano
Departament d'Informàtica de Sistemes i Computadors
Universitat Politècnica de València
Camí de Vera, s/n. 46022 València (Spain)
E-mail: aperles@disca.upv.es

KEYWORDS

Parallel simulation, replication method, coverage analysis

ABSTRACT

Complex discrete-event model simulation requires a lot of computer time in order to get output results statistically valid.

Our research on the area of industrial LANs, storage area networks and real time systems suffer this problem. To reduce this problem, parallel simulation methods have been developed and successfully applied.

The main idea consists of running a set of replications of the same model in parallel, each one driven by a different input set of random numbers. Replications send statistical information of collected output variables to a monitor that decides the end of the simulation when the user specified end condition is achieved.

However, the encouraging results on real complex models needs also a theoretical support to be valuable.

This is the purpose of the work presented here. We evaluate the coverage and performance of our proposals on four queuing models with analytical exact solution.

The results show that the proposed methods provide superior coverage compared with previous methods and a reduction of the bias due to initial transient problem of the replication method.

1. INTRODUCTION AND MOTIVATION

Discrete-event simulation models are necessary for the evaluation and development of technologies susceptible to be outlined using simulation models.

When the simulation model is moderately complex it requires a lot of computational time in order to be evaluated, so many researches have suffered the long response times executing these simulation models.

This is our problem on the research on the area of storage area networks (SAN), industrial LANs and real time systems. The models utilised on these problems try to evaluate an output performance parameter in different scenarios. Figure 1 shows an example of the results evaluating a SAN configuration. Each represented point is the steady-state point estimator \bar{X} of a performance output value with true mean $E[X]$. In order to obtain this graphic we need a repetition of the simulation with different input parameters, and this evaluation can require long execution times. Our purpose is to reduce the execution time of an experiment (set of simulations).

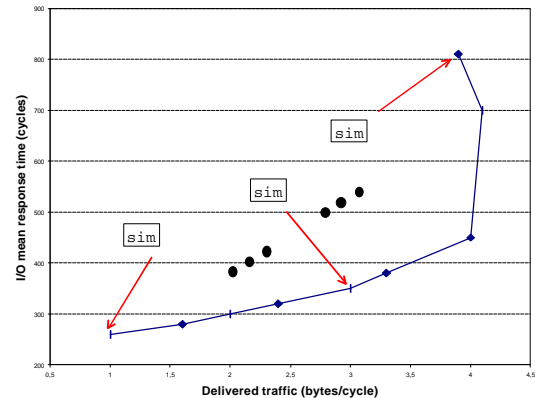


Figure 1. Typical simulation result: "I/O mean response time" versus "Delivered traffic"

In order to improve the response times of simulators many researches are being spent on parallel techniques. There are two main lines:

First, decomposing the simulation model in sub-models that can be executed concurrently in different machines. The main problem of this approximation consists of maintaining the logical consistency of the event list (Fujimoto 1990; Zarei 2000). The use of distributed simulation techniques is not an easy task and requires both a great decomposition effort and model tuning. In addition, a deficient tuning process usually produces poor speed-ups.

It is a good approaching when the purpose is to accelerate the execution of a specific production model. That is, a simulator devoted to be reused many times without changes and changing only the input parameters (i.e. a flight training simulator) and the speed and/or real time is a requirement.

Second, executing copies of the same simulation model in parallel to produce a higher quantity of statistical output data per time unit. This approach could be used for improving response times taking advantage of the "statistical parallelism". This parallelism is well suited to be implemented in a cluster of workstations. Some authors (Ewing et al. 1999; Lin 1994; Rego and Sunderam 1992) proposed tool that attempt to use a group of workstations in order to decrease response time using this approach. We think that this is a good approaching when we want to spend the effort on modelling the system to evaluate and not in adapting the model to a parallel environment. The main idea is to adapt a statistical output analysis method to a parallel environment. See (Yau 1999) for a consideration about which methods seem to be better for this adaptation.

Our research requires the development of lots of simulation models of different nature. Then, our main effort should be this development and not to be concerned in extra efforts on the area of parallelization of models, so the second approach has been taken into account.

In this context, we are working on the development of a tool called Cluster Simulation Experimenter (CSX) (Perles et al. 1999) whose main purpose is to use idle heterogeneous workstations at university laboratories and research centers to run parallel simulations. The CSX design allows its application to any commercial or public simulator, avoiding the apprenticeship of a new simulation language. It provides a general form of monitoring and controlling simulation programs and it has been successfully tested on SMPL (McDougall 1987) and CSIM (CSIM 1998) simulation languages.

This tool has been outlined as a distributed application working under PVM (Geist et al. 1994) that lets the incorporation of Unix and Windows heterogeneous computers to a parallel virtual machine. Basically, parallel simulations executed using CSX are replications of a discrete-event simulation program spawned in different workstations that are monitored in order to extract statistical information of selected output variables.

CSX has been used in our department to improve the long execution time of a simulator of storage area networks (SAN) developed in CSIM. Complex resulting simulation model used previously the batch-means simulation technique. In (Perles et al. 2001) the result of comparing the parallel approach and sequential batch means method is described. We showed that it is better to run a parallel simulation of each point of the experiment than running simultaneously a sequential version for each point in each computer of the cluster.

In this work, our main purpose is to establish the theoretical validity of the method and the performance obtained applying different strategies for the replication management. In order to do that, four queuing simulation models analytically tractable have been elected and tested. Two of these are open system models and the other two are closed system models.

Different factors of the replication methods are evaluated on these models: statistical reset, "warm start" (explained later), sample size, number of replications and termination test from the coverage and performance (needed simulation time) point of view.

The rest of the paper is organized as follows. Section 2 describes our parallel implementation of replications method. Section 3 describes the factors that affect the replications. Section 4, describes open and closed models utilized for this study. Section 5, describes how experiments has been done and the obtained results. Finally, section 6 summarizes the conclusions.

2. PARALLEL REPLICATION IMPLEMENTATION

We can use different statistical output analysis technique in CSX, but our main effort has been on the

independent replication method. See (Law and Kelton 2.000) for a discussion about the election of this method compared with others. This method proposes that we can run k replications of a simulation program using independent input random streams to compute a sample mean.

This is the theoretical support of replication method. Let be \bar{X}_j the sample mean of the observations of some output variable for the j th replication. These observations can be regarded as realizations of independent and normally distributed random variables. Then we can compute an estimator of the true mean $\mu=E(X)$ as,

$$\bar{\bar{X}} = \frac{\sum_{j=1}^k \bar{X}_j}{k}$$

In order to obtain an asymptotically valid $(1-\alpha)100\%$ confidence interval for the grand mean $\bar{\bar{X}}$ we can apply the equation,

$$\bar{\bar{X}} \pm t_{1-\alpha/2, k-1} \sqrt{\frac{S_k^2}{k}}$$

where S_k^2 is the variance of sample means and t is the Student distribution.

The sequential and classic replication method consists of running fixed size simulations and test if they have been reached the wished confidence interval at a given confidence level. If the replications are too short it will be necessary to start another simulation execution of a longer size. On the other hand, if they are too long, the results will be unnecessarily more accurate, wasting computer resources and increasing response time. The difficulty to predict the number of replications and their respective lengths has motivated the development of alternative methods that relax the independence hypothesis, at the same time they are easily implemented in a sequential computer. Some examples are batch-means, spectral and regeneration methods, which add the advantage that only one initial transient period must be eliminated for steady-state simulations.

The problem of the sequential scheme of the replication method can be avoided by using parallel techniques. In this approach we can elect two ways: running fixed size replications in parallel until the wished confidence interval is achieved (Lin 1994) or keep running a fixed and low number of replications until they produce the needed confidence interval.

We have developed this second option because it seems to be statistically more robust on the assumption of independence hypothesis and normal distribution of the output variable. In (Glynn 1995) it is established the asymptotic superiority of a few long replications relative to a large number of shorter replications, assuming that no initial transient deletion is attempted.

So the replication method analyser of CSX basically consists of executing a fixed set of independent replication and let it to run until the wished confidence interval is achieved. Our first implementation let each replication runs until a specified point (number of collected samples or simulated time) send statistical information to a monitor, and wait until the monitor decides if the simulation must continue based on the information sent by the rest of replications (see Figure 2).

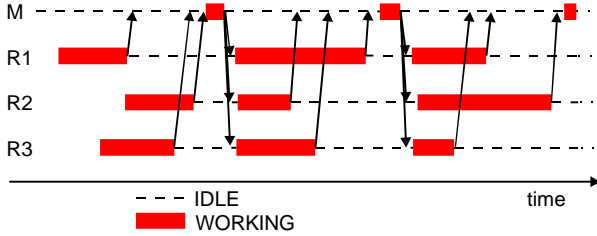


Figure 2. Time scenario of a monitor and 3 replications

Figure 2 shows that the duration of the simulation will be, almost, the sum of the duration of the slower replication in each checkpoint. In (Bhavasara and Isaac 1987) it is established that the execution times of replications are identical independent (i.i.d.) exponentially distributed random variables with mean 1. We solve this problem allowing each replication to run as fast as possible sending, asynchronously, statistical information to the monitor, thus avoiding synchronisation. The monitor can save statistical information and process the next checkpoint when it has information of all replications (see Figure 3).

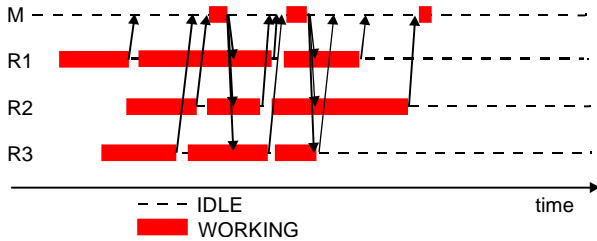


Figure 3. Optimal utilization of computational resources avoiding synchronization effect

3. REPLICATION MANAGEMENT

We assume that we run a fixed number of replications and each replication runs on a different workstation coupled through a local area network. Different parameters of the replication method can be adjusted and they are: statistical reset, "warm start", sample size, number of replications and termination test. These parameters are explained here.

3.1 Sample size

Sample size parameter specifies checkpoints for each replication. A checkpoint implies that each replication sends a sample of collected statistical information to the monitor (sample mean of output variables). Low values of this parameter implies higher communication costs and high values can waste resources.

The sample size parameter is a user-specified parameter that can be: the number of collected observations of an

output variable or an amount of simulated time. The user should select one of the possibilities in function of the nature of the output variable to be analyzed in order to avoid ratio bias effects. For example "response time" of a model usually requires the first kind of sample size whereas "throughput" requires the second one.

3.2 Statistical reset

When the purpose of a discrete-event simulation model is to compute a steady-state point estimator as in our case, the simulation usually starts from artificial or arbitrary initial conditions. An important consequence of this is an initial set of biased observations. This is called "initial warm-up".

In order to reduce this phenomenon, an approach is to discard observations until a given point in simulated time or number of collected observations. This strategy has a main drawback. If we discard too many observations then we waste computer resources and if we discard an excessively low number of observations, then the point estimator may be biased. Figure 4 shows an example of the effect of a reset on a real model, in this case, deleting initial observations we can see a faster evolution to the steady-state value.

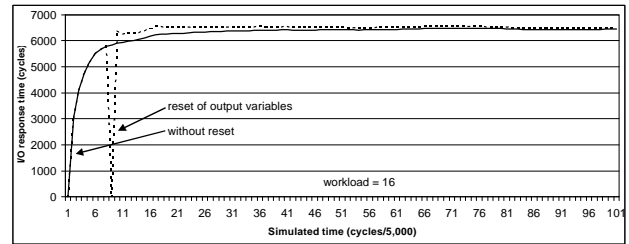


Figure 4. Effect of the reset on the evolution of a point estimator

An important factor evaluated in this work is the effect of discarding some initial observations (called *reset*) on the coverage and performance of the selected models.

3.3 Initial warm-up effect reduction

In (Perles et al 2001) was proposed a method to reduce the effect of the initial conditions of the simulation model.

The literature suggests that it is possible to reduce the effect of the initial transient period (warm-up) by starting the simulation near the steady-state behaviour.

We propose an approaching to this idea. First replication based parallel replication start in the classic empty state. When the replication monitor decides that the end condition has been achieved then it sends model reconfiguration parameters to each running replication in order to continue the simulation of the next point estimator. The result is that the remaining points to be estimated start in the steady state previously achieved, reducing the pernicious effect of the initial empty state.

Initially it seemed us that this approach had an important drawback: the implementation of this strategy. However, our experience showed that it is easy if the implementation of the simulation model takes into account this aspect.

Figure 5 shows an example of the evolution of the point estimator in this strategy.

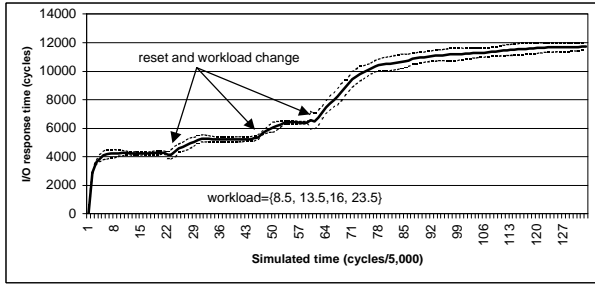


Figure 5. Effect on the point estimator of the strategy to reduce the effect of the initial transient

3.4 Stopping criteria

Once applied the previous strategies, the monitor analyzes the output information in order to decide the end of the simulation. This is called "stopping criteria".

Three stopping criteria based on the precision of the sample mean \bar{X} of the output variable of interest are analyzed:

P1) Relative precision of confidence interval test

A common approach to stop a simulation is to test if the relative precision of the output variable based on the half-width interval of the confidence interval at a given confidence level is achieved. That is,

$$e = \frac{t_{1-\alpha/2, k-1} \sqrt{\frac{S_k^2}{k}}}{\bar{X}} < e_{max}$$

We utilize $e_{max}=0.05$ (5% relative error) at 95% confidence level as the first test for the experiment. We call this P1.

P2) Relative precision and error decrement test

In order to reduce the effect of premature spurious terminations, another common test consists in testing if the relative error is achieved and this error decreases in subsequent samples. We call it P2 and it tests if relative error decreases in the next two samples, that is,

$$e_{max} > e_i > e_{i+1} > e_{i+2}$$

P3) Relative precision and mean slope test

In this test we propose to check if the slope s of the global sample mean \bar{X} between two consecutive samples is near 0 (horizontal) in order to detect the stabilisation of the mean and to avoid some kind of spurious terminations (see Figure 6).

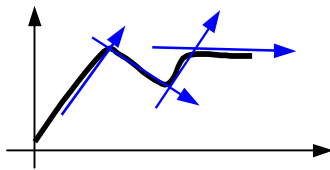


Figure 6. Test P3 attempts to detect slope 0.

This is computed using the expression,

$$\left| \frac{\left(\frac{\bar{X}_i - \bar{X}_{i-1}}{tsim_i - tsim_{i-1}} \right)}{\bar{X}_i} \right| < s$$

where $tsim_i$ is the simulated time at the i -th checkpoint. Or the expression

$$\left| \frac{\left(\frac{\bar{X}_i - \bar{X}_{i-1}}{obs_i - obs_{i-1}} \right)}{\bar{X}_i} \right| < s$$

where obs_i is the number of collected observations at the i -th checkpoint. The election depends on the type of output variable to be analyzed as explained before.

This expression attempts to normalize the analysis in order to be applicable to high and low values of the mean. P3 is defined as,

$$e_{max} > e_i, e_{i+1}, e_{i+2} \quad \text{and} \quad S_{max}=0.0001 > S_i, S_{i+1}, S_{i+2}$$

that is, relative error is lower than e_{max} and slope is lower than S_{max} in the next two samples.

P4) Combination of P2 and P3 tests

In this test we combine both test P2 and P3 in order to decide the termination of the simulation.

To apply this combination both test should hold simultaneously in order to finish.

4. EVALUATED MODELS

In order to evaluate all previously described parameters four models has been elected. Two of these models are open queue network models and two are closed queue network models. The elected simulation models have analytical exact solution to permit the coverage analysis.

Output variable analyzed has been "mean customer response time". All services and input load are exponentially distributed.

The first model is an open M/M/1 queue. Figure 7 shows a queue diagram of this model and Table 1 shows analytical results for "response time" output variable for different values of normalized load.

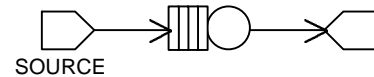


Figure 7. M/M/1 model

Table 1. Analytic results for M/M/1 queue

Normalized load	Mean response time
0.1	1.111
0.4	1.667
0.7	3.333
0.9	10.00

The second model is an open central server model (see Figure 8). Table 2 shows analytical results for different values of input load.

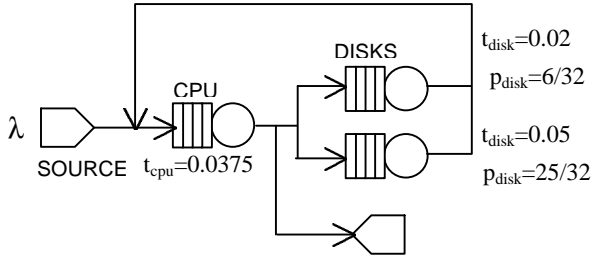


Figure 8. Open central server model

Table 2. Analytic results for open central server model

λ	Mean response time
0.1	2.199
0.3	2.793
0.6	5.366
0.75	13.19

The third model is a closed M/M/3//k queue (see Figure 9). Table 3 shows analytical results for different values of input load.

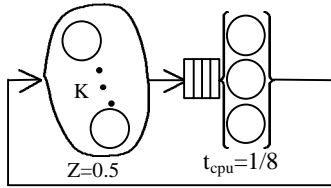


Figure 9. M/M/3//k model

Table 3. Analytic results for M/M/3//5 model

k	Mean response time
5	0.1262
15	0.2108
25	0.5427

The fourth model is a closed central server model (see Figure 10). Table 4 shows analytical results for different number of terminals.

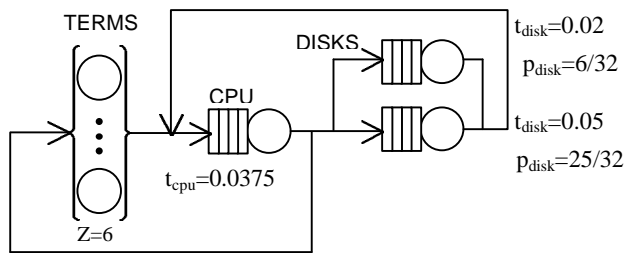


Figure 10. Closed central server model

Table 4. Analytic results for closed central server model

N	Mean response time
1	2.000
4	2.814
8	4.917
12	8.606
100	114.0

5. EXPERIMENTAL RESULTS

To drive the experience, a simulation program equivalent to each model has been written in the CSIM simulation language. As we do not know the nature of the random number generator of CSIM, we decided to change it for the *mrg32k5a* generator (L'Ecuyer 1999) with period 2^{319} to generate independent random streams spaced 10^8 random numbers departing from the original set of seeds (see Figure 11).

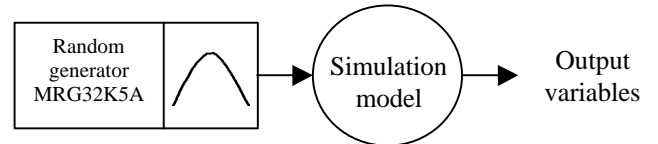


Figure 11. External random number generator drives the model

The experiments have been done for $k = \{3, 5, 7, 8, 10, 15, 20, 30\}$ replications, stopping criteria P1, P2, P3 and P4, and different combinations of sample size, reset and warm-up avoidance.

Coverage of true mean \hat{p} is estimated as the ratio,

$$\hat{p} = \frac{x}{n}$$

where n is the number of simulation experiments and x is the number of experiments that produces a confidence interval for the output sample mean that includes the true mean μ analytically obtained.

The models have been simulated in order to estimate the output variable "mean customer response time". In order to establish checkpoints, the number of collected observations of this variable has been used for the "sample size" parameter.

In order to compute coverage and average simulation cost, each combination of parameters has been repeated 1000 times using independent random streams.

Appendix A and B summarizes graphically a selection of the results of the experiment. Each row has three graphics, the left one represents the achieved coverage based on the number of replications, where a 95% coverage is the theoretical value. The next one represents the average cost of the parallel simulation measured in number of observations in a replication; this value is directly proportional to the required simulation time or the cpu time. And the graphic on the right represents the average total amount of observations required, that is, the sum of the

number of collected observations of each replication in a parallel experiment.

All configurations of replication management provided satisfactory values of coverage for models with low values of load (see figure A-1). Very low loaded systems produced poor speed-up results due to an excess of samples collected in each checkpoint. This is an expected result. For example, if the checkpoint is every S observations, the analysis for 3 replications is done every $3 \times S$ samples and for 30 replications every $30 \times S$. This effect disappears when the load of the model increases and reappears when checkpoint size increments.

When models are heavily loaded, coverage problems appear. Classical termination method P1 produces the worst results and enhances notably when warm-up avoidance techniques are applied.

This analysis shows that open and closed models behave in a different way.

Coverage of M/M/1 open model has low enhancement when initial observations are discarded or warm start is applied. This result has a theoretical explanation because empty state of an M/M/1 is a valid state, although independence hypothesis of replication is relaxed because all models depart from the same state.

Open central server offers similar results. Discarding initial observations or warm-start slightly enhances coverage without affecting simulation cost. In order to achieve a convenient coverage it is necessary a higher checkpoint value.

Closed models exhibit a different behaviour. Methods P1 and P2 show a low coverage when the number of replications increase if no reset of pre warm is applied. This result is because the initial empty state is not a natural model state and the samples collected in this phase are highly biased. P3 test provides the best behaviour in contrast of the results of open models. The nature of closed models affects the parallel performance of the replication method, limiting the parallel speed-up. In this case is true the literature proposal: it is better a low number of long replications than a high number of short replication. Discarding initial observations and applying prewarm we can reduce this nocive result from a parallel point of view (see figure B-5, and B-6).

6. CONCLUSIONS

The results obtained in this work show that the stop criterion P1 widely applied in sequential simulation studies can provide low coverage results. This problem is particularly important on models where the initial transient response is biased.

P2 stopping criteria is the best choice for selected open models and P3 is better for loaded closed models that provide. As expected, P3 is better for models that exhibit a slow tendency to the true mean. In order to get the best of both stopping criteria, the combination of P2 and P3 made in P4 provide an interesting stopping criteria for all the models tested.

Combining both initial observations discard and warm start it is possible to increment the number of replications in parallel maintaining convenient coverage.

An adequate checkpoint size, stopping criteria P4, discarding a low number of initial observation and warming the model with the previous simulation provides in combination with the parallel replication manager designated provides an excelent tool to decrease the response time of real complex discrete-event simulation models. This has been experienced with a real complex SAN simulation model.

7. REFERENCES

- User's guide: CSIM18 Simulation Engine (C version). *Mesquite Software, Inc.* 1998
- Ewing G., Pawlikowski K. and McNickel D.. Akaroa2: Exploiting Network Computing by Distributing stochastic Simulation. *Proceedings of the 13th European Simulation Multiconference (ESM'99)*. May 1999.
- Fujimoto R.M.. Parallel discrete event simulation. *Communications of the ACM*. No. 10. October 1990.
- Geist A., Beguelin A., Dongarra J., Jiang J., Manchek R. and Sunderam V. PVM3 user's guide and reference manual, Technical Report ORNL/TM-12187. Oak Ridge National Laboratory, May, 1994.
- Glynn P.W. Some new results on the initial transient problem. *Proceedings of the Winter Simulation Conference*. 1.995
- Law A. M., Kelton W. D. *Simulation Modeling and Analysis*. Ed. McGraw-Hill. 2.000
- L'Ecuyer P. Some recommendable uniform random number generators. *Proceedings of the 13th European Simulation Multiconference (ESM'99)*. May 1999.
- Lin Y.-B. Parallel Independent Replicated Simulation on a Network of Workstations. *Proceedings of the 8th Workshop on Parallel and Distributed Simulation (PADS'94)*. Vol. 24. No. 1. July 1994.
- McDougall M. H.. *Simulating Computer Systems*. The MIT Press, Cambridge, Massachusetts. 1987.
- Perles A., Martí A., Serrano J.J.. Clustered Simulation Experimenter: A tool for concurrent simulation on loosely coupled workstations. *Proceedings of the 13th European Simulation Multiconference (ESM'99)*. May 1999.
- Perles A., Molero X., Martí A., Santonja V., Serrano J.J. Improving the Execution of Groups of Simulations on a Cluster of Workstations and its Application to Storage Area Networks. *Proceedings of the 34th Annual Simulation Symposium (SS'2001)*. April 2001.
- Rego V.J. and Sunderam V.S.. Experiments in Concurrent Stochastic Simulation: The EclIPSe Paradigm. *Journal of Parallel and Distributed Computing*. 14, 66-84. 1992.
- Yau, V. Automating parallel simulation using parallel time streams. *ACM Transactions on Modeling and Computer Simulation*. vol. 9. n. 2. pp:171-201. April 1.999
- Zarei B. A taxonomic review of parallel discrete event simulation. *Proceedings of the 12th European Simulation Symposium (ESS'00)*. September 2000.

APPENDIX A. Open models coverage and performance results. Output variable R_{system}

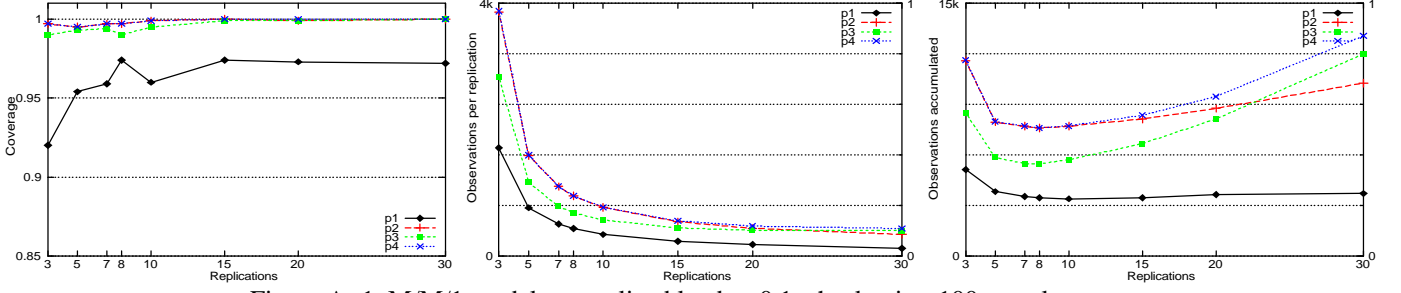


Figure A- 1. M/M/1 model, normalized load $\rho=0.1$, checkpoint=100 samples, no reset.

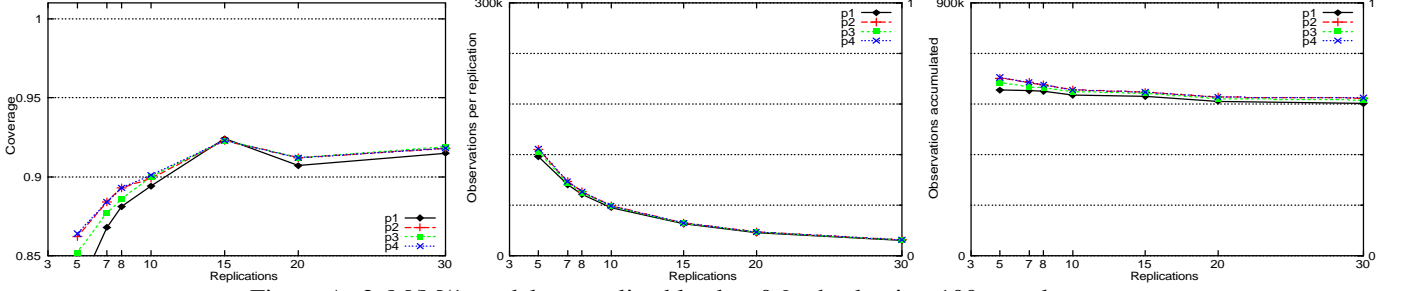


Figure A- 2. M/M/1 model, normalized load $\rho=0.9$, checkpoint=100 samples, no reset.

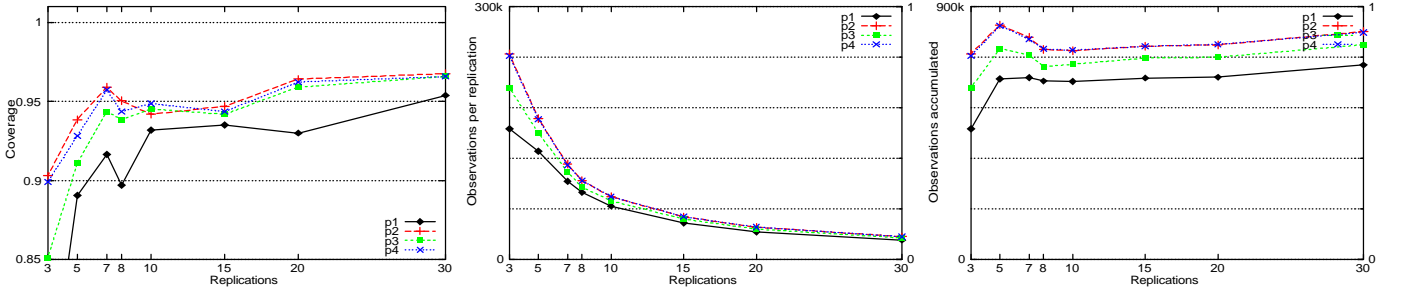


Figure A- 3. M/M/1 model, $\rho=0.9$, checkpoint=1000, reset=4000, warm start.

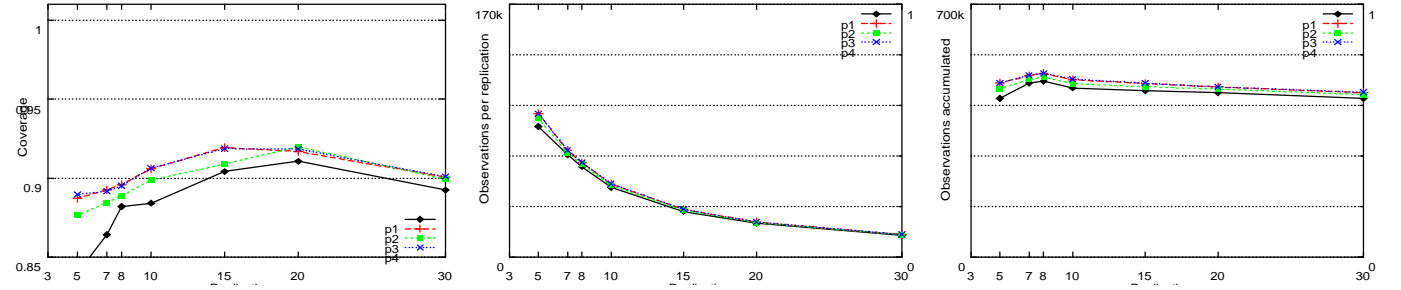


Figure A- 4. Open central server, $\lambda=0.75$, checkpoint=100, no reset.

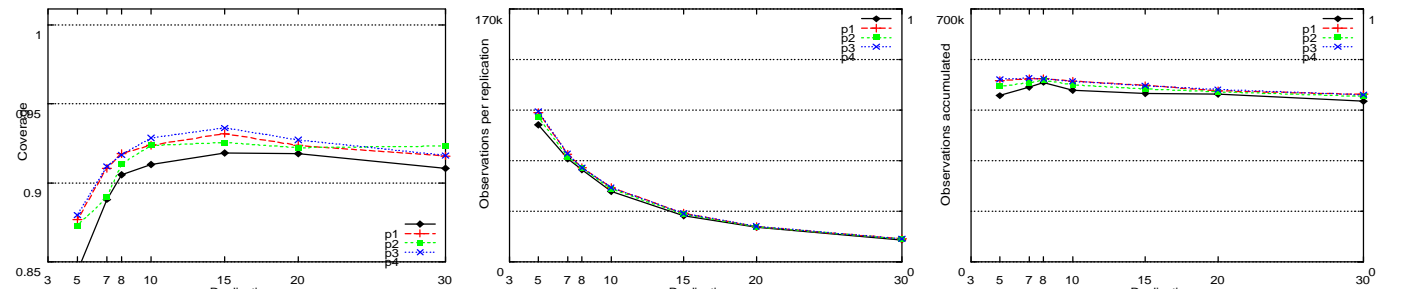


Figure A- 5. Open central server, $\lambda=0.75$, checkpoint=100, no reset, warm start.

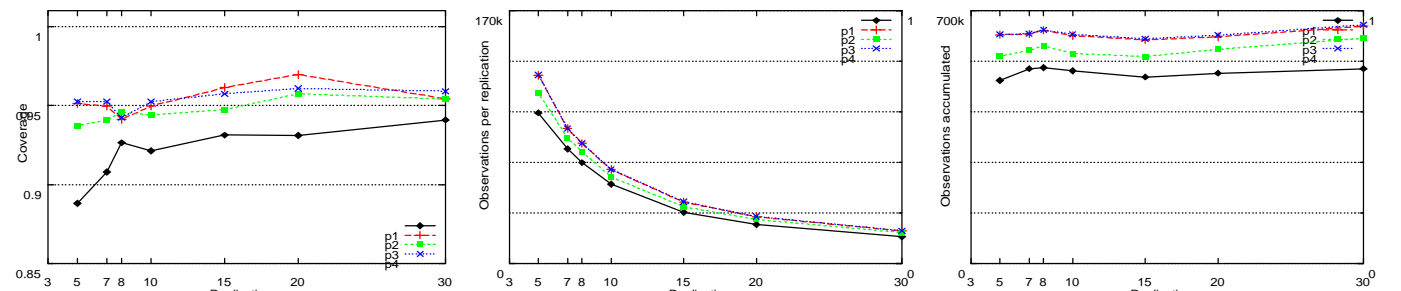


Figure A- 6. Open central server, $\lambda=0.75$, checkpoint=1000, reset=2000, warm start.

APPENDIX B. Closed models coverage and performance results. Output variable R_{system}

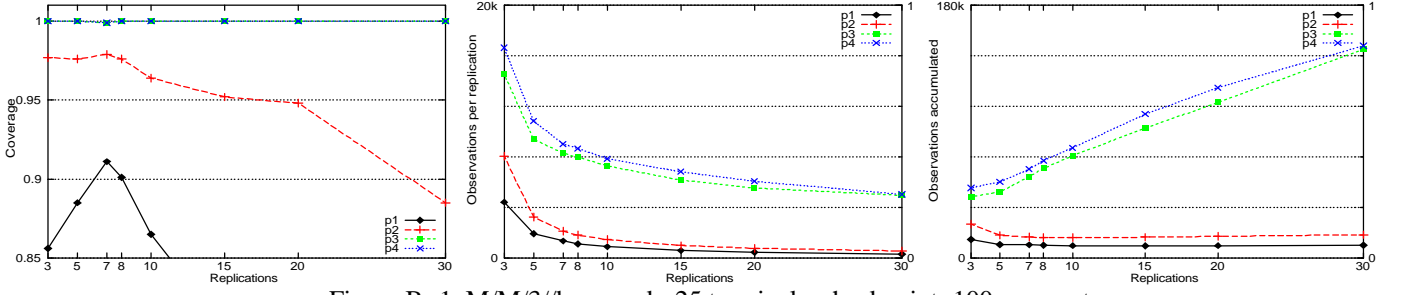


Figure B- 1. M/M/3/k queue k=25 terminals, checkpoint=100, no reset.

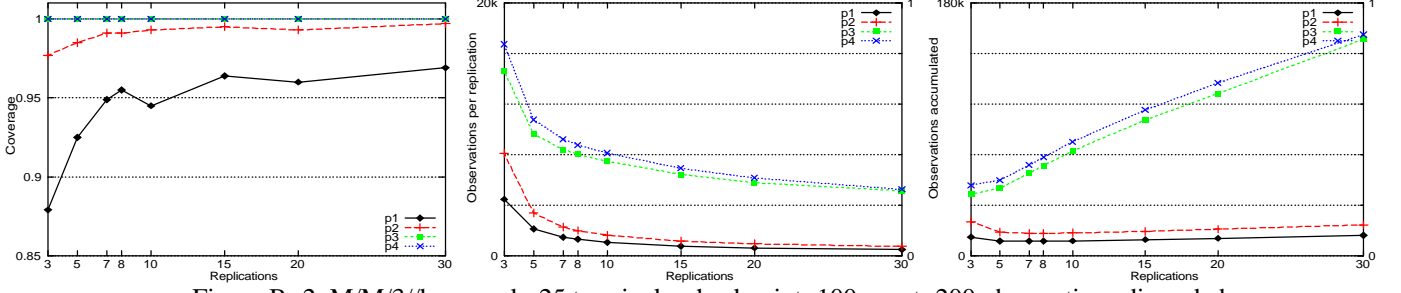


Figure B- 2. M/M/3/k queue k=25 terminals, checkpoint=100, reset=200 observations discarded.

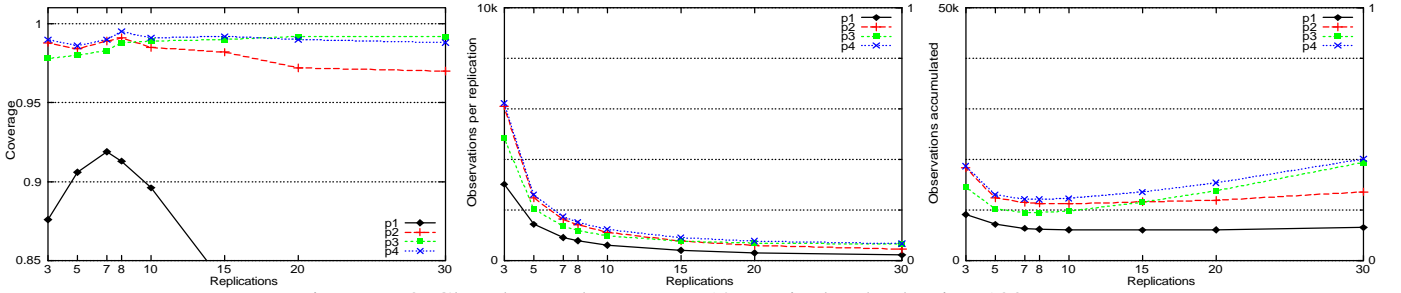


Figure B- 3. Closed central server, N=12 terminals, checkpoint=100, no reset.

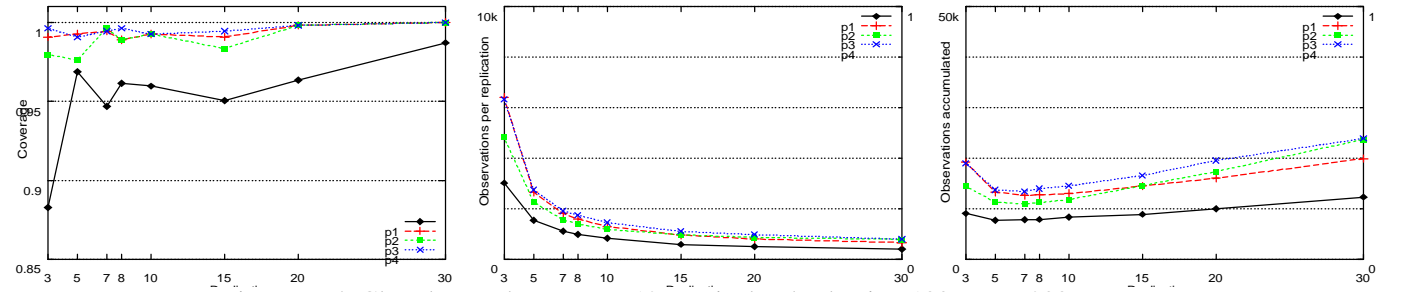


Figure B- 4. Closed central server, N=12 terminals, checkpoint=100, reset=200, warm start.

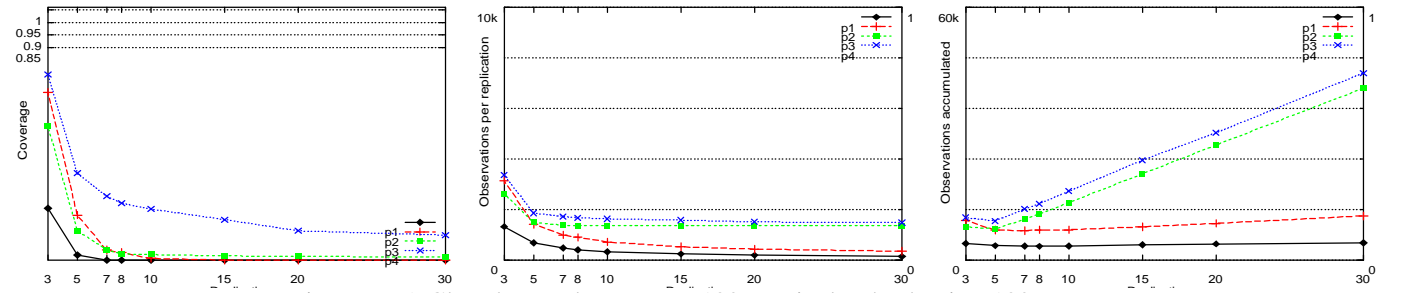


Figure B- 5. Closed central server, N=100 terminals, checkpoint=100, no reset.

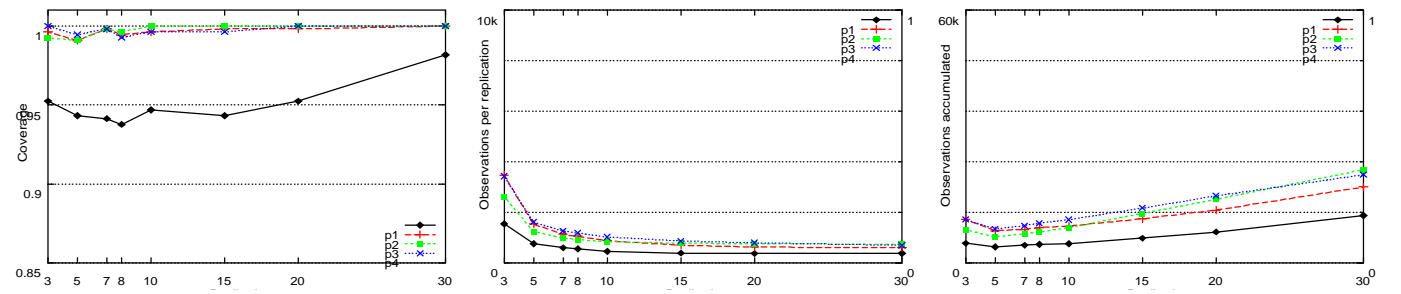


Figure B- 6. Closed central server, N=100 terminals, checkpoint=100, reset=200, warm start.

Analysis in Management

SETTING AND ANALYSIS OF MULTIPRODUCT SUPPLIER LINKS MANAGED BY KANBAN USING A LEARNING + EVOLUTIONARY OPTIMIZATION

Anne-Lise Huyet

Jean-Luc Paris

LIMOS, UMR CNRS 6158

I.F.M.A. - B.P. 265, F-63175 Aubière Cedex

Phone: (33) 73-28-80-00, Fax: (33) 73-28-81-00

E-mail: alhuyet@ifma.fr, paris@ifma.fr

Abstract.

The problem addressed in this article is not only to optimize the configuration of manufacturing system including supplier links. It is also to contribute to analyze its behavior through highlighting relations between parameters of the system and their influence on its performance. The proposed approach is based on evolutionary optimization simulation approach working in synergy with a learning approach. The learning approach guides the search for solutions by giving characteristics of good solutions to the evolutionary optimization simulation approach. And this one orientates the learning approach on promising areas. It is illustrated through the setting and the analysis of a multiproduct kanban supplier links.

INTRODUCTION

Currently, a large number of manufacturing firms are seeking methods that can help them to identify and implement strategies for analyzing and improving their supply chain networks. Nevertheless, decision making in supply chains is a complex process due to the large scale of the networks or to the interactions among supply chain elements. Supply chain decisions have been classified into three categories: strategic (long-term), tactical (medium-term), and operational (short-term and real-time) according to the time horizon of the decisions (Biswas and Narahari 2003). Supply chain performance measures can be classified broadly into two categories (Viswanadham 1999): qualitative measures (such as customer satisfaction or product quality) and quantitative (such as order-to-delivery lead time, resource utilization, work-in-progress...). In our case, we will deal with an operational level. Operational supply chain decisions include the determination of the best value of several parameters such as the production load units or the number of kanban for the production line and such as the transport load units for the supplier links. The parameters have a real impact on the system's performance that can be expected. Unfortunately, they strongly interact. These interactions make the setting and the analysis of the system extremely complex and it turns out to be very difficult to solve analytically, unless restrictive assumptions are formulated. Simulation may be used to evaluate the performance criterion to optimize (Gaury et al. 1997). The difficulties are then to know how to drive the simulation experiments in order to determine

the value of each parameter and how to exploit those experiments in order to analyze the system.

Optimizing the parameters of manufacturing systems has already been treated by the 'optimization simulation' methods as gradient method, stochastic methods, response surface and heuristics methods (Azadivar 1992). Nevertheless, all those approaches work like black boxes with the best solution as only output. But, with the actual economic investments, the best found solution is not enough for the managers; they also need an analysis of the behavior of the system and alternative solutions, which can also improve the performance of the system.

Extracting information on the system behavior in order to analyze it, has also been explored through different approaches of machine learning or data mining. Unfortunately, the difficulty to build the learning set remains and has an important impact on the relevance of the results.

In this article, the authors will set up and jointly analyze a multiproduct kanban supplier links using intelligent optimization, where an optimization approach works in synergy with a learning approach.

PROBLEM DESCRIPTION

The supply chain management takes into account not only the production line design or configuration but, also the customer/supplier relationship. Nevertheless, the setting of the production policy and the transport policy are very often considered separately in the literature. Since all these issues pertain to the same manufacturing system, it would be logical if they are optimized and analyzed together (Sarker and Balan 1999) as it will be done here.

In this article, the system has a just-in-time (JIT) production policy with production load units and a transport policy with transport load units. In a production line operating under a JIT production policy, the output rate of the last machine is generally regulated by the products needs. This technology is often implemented both to improve the productivity of the line and to reduce the manufacturing costs (Golhar and Stamm 1991, Price et al. 1994, Yavuz and Satir 1995, Huang and Kusiak 1996).

The system studied here, is constituted of a buyer group and a supplier which can produce three types of products. The relationship is done by a lorry which brings the orders of

the buyer group to the supplier and delivers the ordered products to the buyer group. We suppose that the buyer group orders 65% of product 1, 25% of product 2 and 10% of product 3. The orders and the ordered products are managed by transport load units. The supplier manufacturing system is constituted of a production line which carry out three processes. There is a stock that we assume infinite and a fixed number of kanban by type. A schema of this system is presented in fig.1.

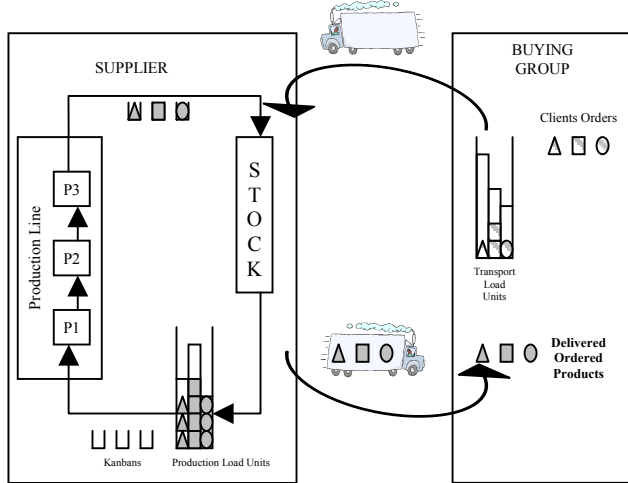


Fig. 1: Studied System

In our example, the problem to address is to determine values for the following parameters:

- KB_i : number of kanbans in front of the production line for the product of type i , 3 variables. We consider that the number of kanbans can vary between 10 and 30.
- TLU_i : transport load units associated to product i . We consider that transport load units can take value between 1 and 20.
- PLU_i : production load units associated to product i . We consider that production load units can take value between 1 and 10.

These three types of parameters compose a chromosome of 9 genes $X = [K_i, T_i, P_i]$ with $i = 1..$ number of product types. For economical reasons, we have to minimize the order-to-delivery lead time as the number of WIPs. The objective function to maximize can be expressed as follows:

$$F = M - (C_{WIP} * \text{number of WIPs} + \sum C_i * \text{satisfaction delay for order of product } i), \text{ with the given constant values:}$$

- M : large constant,
- C_{WIP} : cost of one WIP,
- C_i : cost of one unity of time for orders delay of product i ,
- i : number of type of product

LEARNING + EVOLUTIONARY OPTIMIZATION

First the principle of intelligent optimization will be presented. Then, with reference to the fig. 2, the different modules of the method will be described as well as how those different modules are connected.

Method Principle

In the current industrial context, the decision makers cannot be satisfied with an adjustment of the level flow, even optimum, of their system of production without any detail on its critical points or its behavior. Thus we propose a methodology which will provide, at the same time, performing configurations of the system and knowledge on its parameters, such as combinations to be supported or avoid and a hierarchy of parameters according to their impact on the system performance. This methodology, which can be qualified of "intelligent optimization", is based on the work in synergy between an evolutionary optimization approach, for the performance aspect, and a learning approach, for the knowledge aspect.

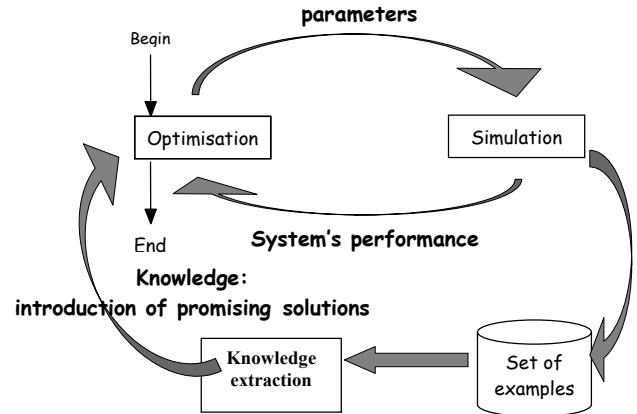


Fig. 2: "Intelligent Optimization"

This methodology will allow exploiting the advantages of the two approaches while filling their gaps. Thus, one of the major problem involved in evolutionary optimization is the number of solutions to be evaluated, expensive problem in computing time especially when the evaluation is done by simulation, as in this case. The learning approach will direct search towards promising areas of solutions space and maintain enough diversity to avoid the local optima. Reciprocally, one of the critical points of the learning approach is the construction of the data set on which knowledge will be extracted. Evolutionary optimization will automatically generate sets of data centered on what interest the managers, which are the areas of the solutions space where the system has a high performance. Our methodology is described in fig. 2.

The evolutionary optimization module generates various combinations of parameters in order to constitute its population. The module of simulation evaluates these individuals. Evaluations are taken into account by the optimization module, which generates a new population. In our approach, we add 3 stages to this traditional optimization via simulation method. Firstly, we collect the examples generated by the optimization module and evaluated by simulation. Secondly, we apply a learning approach to this collection and, thirdly, we exploit the knowledge extracted by transmitting it to the optimization module: new individuals generated according to this knowledge are introduced in the current population. Thus,

the set on which learns the learning module is not randomly generated in the global search space. But it is focused on interesting areas and it contains increasingly performing examples generated by the evolutionary optimization module. The characterization of the good solutions will bring new knowledge on the system, which is introduced into the optimization module to guide it towards the areas containing of the promising solutions.

Simulation Optimization

The first part of the problem to address is to determine the best combination of variables (e.g. transport load units, production load units, number of kanbans) in order to optimize an objective function, such as the total time needed to complete N products (makespan) or the production cost. Several simulation optimization methods have already been used in the manufacturing system area, for example, response surface, direct search and perturbation analysis (Azadivar 1992). Among these methods, evolutionary algorithms (EAs) are really promising (Tautou and Pierreval 1995), (Dolgui et al. 2000), (Fontanili et al. 2000), (Paris and Pierreval 2001) and generate sets of evaluated examples, which remain almost unexploited.

Evolutionary algorithms include various methods, such as genetic algorithms, evolution strategies, evolutionary programming, genetic programming, etc (Bäck 1996). These approaches have in common the principles inspired by natural evolution phenomena. One of its essential principles is that a strong (well adapted) individual has a high probability to survive.

EAs make the initial population evolve toward a population, which is expected to contain the best solution. For that, they use the following *reproduction-evaluation* cycle: for each iteration, called *generation*, chromosomes from the current population are selected with a given probability and copies from these individuals are created. Selection of individuals is based on their fitness relative to the current population, i.e. the strongest individuals will have a higher probability to be copied. In a simulation optimization approach by evolutionary algorithm, an individual is a vector of input values of a simulation model and its *fitness* is a function of the model's output variables. New individuals, produced by this process, are submitted to the *mutation* and the *recombination operator*. The recombination mechanism allows the mixing of parental information while passing it to their descendants (offspring) and the mutation introduces innovation into the population. From a generation to another, individuals have globally a higher fitness. For our purpose, we will use an elitist EAs using the principle of the wheel, with a normal law mutation and a uniform recombination. The general frame of the evolutionary algorithm that has been implemented is described in fig. 3.

- | | |
|---------|--------------------------------------------------------------------|
| Step 1: | Start with the generation counter equal to zero ($t=0$). |
| Step 2: | Initialize a population of individuals $P(t)$. |
| Step 3: | Evaluate fitness of all initial individuals in population $P(t)$. |
| Step 4: | Select sub population to be submitted to variation |

- | | |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------|
| | (selection). |
| Step 5: | Apply variation operators to selected parents (recombination and/ or mutation).
$P'(t) = \text{Variation}[\text{selection}\{P(t)\}]$ |
| Step 6: | Evaluate the mated population's fitness (evaluation). |
| Step 7: | Select the survivors from actual fitness (survival). |
| Step 8: | $P(t+1) = \text{select}[P'(t)Q]$. |
| Step 9: | Increase generation counter ($t=t+1$). |
| Step 10: | Test for termination criterion (number of generations, fitness, etc.) and stop or go to step 4. |

Fig. 3: Evolutionary algorithm (Bäck et al. 1997)

Although EAs are known to be a powerful optimization technique, they are notoriously slow and work as a “black box” so they give no explanation on the system behavior or no alternative solutions. This is why, with our objective, the simulation optimization approach needs to be combined to a learning approach.

Learning Approach

Our methodology requires a learning approach, which will allow us not only to induce an identification process of a solution class (in our case, relative to a level of performance), but also to induce a hierarchy of the parameters according to their criticality and their impact on the system performance. There are many methods of machine learning or data mining (Michalski et al. 1984). Among them, the networks of neurons (Bishop 1995, Mitchell 1997) or Genetic Based Machine Learning (Mitchell 1997, Bonnarini 1997)) are used in various approaches but still remain ‘black boxes’. Nevertheless, the family of the induction graphs (Zighed and Rakotomalala 2000) is able to provide a hierarchy of parameters as a learning process. Moreover, the procedure of construction is comprehensible by the various people concerned with the problem and the induction graphs representation makes easier the communication of the results. To implement our learning module, we have chosen the C4.5 algorithm developed by (Quilan 1993). The goal of this type of approach is to organize the parameters into a tree in which the more an attribute is relevant, the more it is close to the root. The relevance of an attribute is its capacity to classify an example. Each node of the tree - except the sheets - corresponds to a parameter; each branch of a node corresponds to a value of this parameter. The relevance of a parameter according to a given data set depends on its capacity to separate the data of this set in the most homogeneous possible subsets (measurement from its entropy). A data set is associated to each node: the initial set for the root and, for the other nodes, the set resulting from various separations according to the way (succession of parameters and values of these parameters) from the root to the concerned node. A set containing only data relative to the same class is associated to each sheet. Nevertheless, whatever efficient is the learning method; it is strongly dependent of the relevance of the training. The learning set must contain enough representative examples if we want to obtain a good prediction for the non classified examples but how can we determine which examples are representative? It is strongly dependant of the problematic. In our

problematic, we are interested in good examples. But we want to explore all the regions of solutions space containing good examples for two reasons. First, the research could fall in a local optimum and stay in its neighborhood. In this case, no more knowledge will be achieved. Secondly, we still have the objective of providing different ways of improvement for the system. In this context of manufacturing system, an optimal solution is not always needed or not always usable at this time so, managers could prefer a near optimal solution or simply a better than the current one. This is why, enough diversity in the good solutions set must be maintained.

Method Implementation

The connection of the optimization module to the learning module will be simply done by delivering to the learning module some examples already used and evaluated in the optimization phase. We chose to carry out this delivery every 3 generations, and it is the data set generated since the previous delivery, which will be delivered. Other researches will concentrate on the influence of the frequencies and volumes of delivery. It should be noted that, progressively, the learning data sets will contain more examples from the areas that interest the manager, namely the most promising areas. The connection of the learning module to the optimization module will consist in injecting into the population a certain number of promising solutions built from the induction graph.

With reference to the induction graph, it will be scan in order to achieve sheets of very good class (coded by 1). The characteristics of these promising solutions are described by the attributes and the values recorded on the ways between the root and the sheets. This way defines a profile from which new solutions, that we hope very good, will be built. Such solutions will be built by freezing the way parameters to the recorded values and by randomly generating the other values. For example, with reference to fig. 5, a promising solution could be a solution where parameters TLU1, PLU3, KB2 and KB1 are respectively set to 7, 3, 10 and 14 and others parameters are submitted to random choice. In order to keep a constant population size, after the injection of new individuals, we use a mechanism of selection in

order to eliminate the worst individuals. Adding a new step to the evolutionary algorithm previously described concretizes these connections:

- Step 9': if (condition to call learning)
1. Construct the induction graph.
 2. Characterize good regions.
 3. Generate typical new individuals of those regions.
 4. Evaluate them.
 5. Integrate them in the current population.

Fig. 4: New step added to EA

APPLICATION CASE STUDY AND RESULTS

The evaluation of F is performed with a simulation model. For the optimization module, the size of the population will be of 30 individuals and the maximum number of generations will be equal to 200, these values have been determined by a study of the convergence of the results. The parameters of the EA have also been studied and for this example, better results have been obtained with a crossover probability of 0.3, a mutation probability of 0.8 and an elitist ratio of 10. Single-point crossover and uniform mutation are used (Tautou and Pierreval, 1995). The induction graph learning has been called every 3 generations of the EA. The learning set will have a size of 90 examples, which correspond to the last 3 populations used by the EA. The maximum number of characterized good regions has been fixed to 10.

First, we have implemented optimization via simulation using only our basic EA in order to have a reference value for the fitness. Then, we have implemented our approach. The better solutions were quite similar (only 4% more efficient with Intelligent Optimization).

This is in term of learning results that we have the major contributions. There are three main aspects. First, we have a hierarchy of the parameters regarding to their impact on the performance, as it is represents in fig. 5. For this example, the more important parameter is the Transport Load Units of product 1.

Secondly, we have indications on which combinations of parameters favor and which must be avoid. If we choose

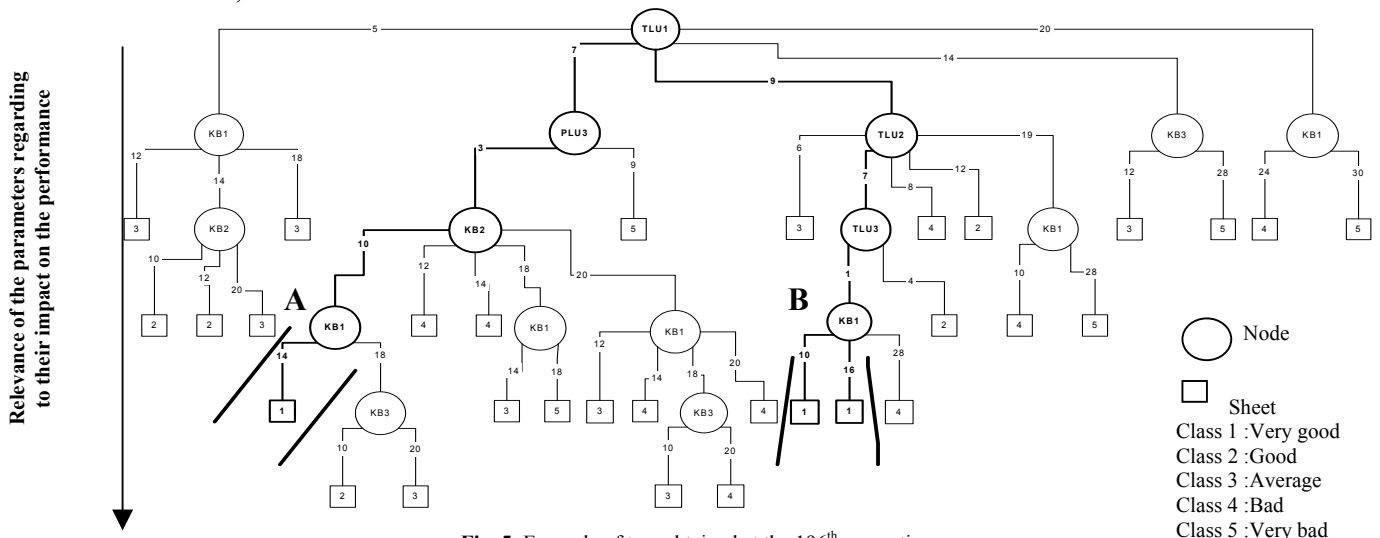


Fig. 5: Example of tree obtained at the 196th generation

Transport Load Units of capacity 7 for product 1, Production Load Units of capacity 3 for product 3, 10 kanbans for the product 2 and 14 kanbans for the product 1, it will be interesting because it induces solutions of class 1. On the other hand, if we choose Transport Load Units of capacity 20 for product 1 and 20 kanbans for the product 1, it will generate worse solutions. With those indications, we can construct profiles of good solutions that are more and more relevant. A kind of skeleton of how a good solution looks like can be given to the managers. Then, they have more flexibility in their choice and can focus investments on relevant modifications.

Thirdly, we can analyze the “breakings” of the tree in order to better understand the complexity of the system. We call “breakings”, the nodes where at least one sheet is classified very good and the others have worse performance. On the example presented in fig. 5, two breakings can be identified with a letter. For each breaking, we want to explore the neighbor branches in order to understand what is happened with the other values. For example, with (A), why less or more than 14 kanbans for product 1 seems provide worse performance? Then we have tested a configuration based on the first bold way but with only 10 kanbans for product 1. It appears that all the order-to-deliver lead times are higher (especially for product 1). But the inventory is less important than with 14 kanban for product 1. On the other hand, a solution with 16 kanbans for product 1 will decrease the order-to-deliver lead times but increase significantly the inventory of WIP. For the second breaking (B), if 8 kanbans for product 1 are considered, the orders are not anymore satisfied in an acceptable way. If this parameter is fixed at 18, the orders are satisfied in the same delay than with 16 but the inventory has increased. Then, it is not interesting to set up less than 10 or more than 16 kanbans for product 1 when TLU1 is 9, TLU2 is 7 and TLU3 is 1.

CONCLUSION

In this article, we have not only optimize the configuration of manufacturing system including supplier links but, also contribute to analyze its behavior through highlighting relations between parameters of the system and their influence on its performance. First, we have a hierarchy of the parameters regarding to their impact on the system performance. Secondly, we have indications on which combinations of parameters favor and which must be avoid. And thirdly, we achieve a better understanding of the system behavior by looking more precisely what's happen in the neighborhood of our better solutions.

The proposed approach of Intelligent Optimization is based on evolutionary optimization simulation approach coupled to a learning approach. The learning approach guides the search for solutions by giving characteristics of good solutions to the evolutionary optimization simulation approach. And this one orientates the learning approach on promising areas.

Although the approach presented in this paper was illustrated using the setting and the analysis of a multiproduct kanban supplier links, it is well suited to different application areas. It should therefore be useful to solve a wide range of simulation optimization problems especially when the system is complex and the evaluation of a solution is time-consuming.

REFERENCES

- Azadivar, F. 1992. A tutorial on simulation optimization, *Proceedings of the 1992 Winter Simulation Conference*, 198 - 204.
- Bäck, T. 1996. *Evolutionary Algorithms in Theory and Practice*, Oxford University Press.
- Bäck, T.; U. Hammel; and H.P. Schwefel. 1997. Evolutionary Computation: Comments on the History and Current State, *IEEE Transaction on Evolutionary Computation*, Vol. 1, n° 1, pp. 3-17.
- Bishop, C. 1995. *Neural Networks for Pattern Recognition*, Oxford University Press.
- Biswas, S. and Y. Narahari 2003. Object oriented modelling and decision support for supply chains, *European Journal of Operational Research*, In Press.
- Bonnarini, A. 1997. Evolutionary learning of fuzzy rules : competition and cooperation, *Fuzzy Modelling: Paradigms and Practice*. Kluwer Academic Publishers, Norwell, MA.
- Dolgui, A.; A. Ereemev; A. Kolokolov; and V. Sigaev. 2000. A Genetic Algorithm for Buffer Allocation in Production Line with Unreliable Machines *Proceedings of the International Workshop on Discrete Optimization methods in Scheduling and Computer-Aided Design*, Minsk, September 4-6, 26-32.
- Fontanili, F.; A. Vincent; and R. Ponsonnet. 2000. Flow simulation and genetic algorithm as optimization tools, *International Journal of Production Economics*, Vol. 64, 91-100.
- Gaury, E.G.A.; J.P.C. Kleijnen; and H. Pierreval. 1997. Configuring a pull production control strategy through a generic model, *Proceedings of the conference on management and control of production and logistics*, Campinas, Brazil, September, pp. 223-228.
- Golhar, D.Y. and L.C. Stamm. 1991. The just-in-time philosophy: a literature review, *International Journal of Production Research*, 29 (4), 657-676.
- Huang, C.C. and A. Kusiak. 1996. Overview of Kanban systems, *International Journal of Computer Integrated Manufacturing*, 9, (3), pp 169-189.
- Michalski, R.S.; J. Carbonell; and T. Mitchell. 1984. *Machine Learning*, Tioga Press, Palto Alto., Calif., USA.
- Mitchell, T. 1997. *Machine Learning*, Mc-Graw-Hill.
- Paris, J.L. and H. Pierreval. 2001. "Dealing with design options in the optimization of manufacturing systems: an evolutionary approach", *International Journal of Production Research*, vol. 39, n°6, pp 1081-1094.
- Price, W.; M. Gravel and A.L. Nsakanda. 1994. A review of optimisation models of Kanban based production systems, *European Journal of Operational Research*, 75, pp 1-12.
- Quinlan, J.R. 1993. *C4.5: Programs for Machine Learning*, Morgan Kaufmann.
- Sarker, B.R. and C.V. Balan 1999. Operations planning for a multi-stage kanban system, *European Journal of Operational Research*, vol 112, pp. 284-303
- Tautou, L. and H. Pierreval. 1995. Using evolutionary algorithm and simulation to optimize manufacturing systems, *Proceedings of the conference INRIA/IEEE ETFA'95*, October 10-13, Paris (France), Vol. 3, pp 509-516.
- Yavuz, I.H. and A. Satir. 1995. A kanban-based simulation study of a mixed model just-in-time manufacturing line. *International Journal of Production Research*, 4,(33),
- Zighed, D.A. and R. Rakotomalala. 2000. *Graphes d'induction*, Hermes.

SIMULTANEOUS PERTURBATION STOCHASTIC APPROXIMATION TO OPTIMIZE DRUG MANAGEMENT IN HOSPITAL SYSTEMS

M. Albanese, A. Matta and T. Tolio

Dipartimento di Meccanica

Politecnico di Milano

Milano,

Italy

E-mail: albanese@tmhp1.mecc.polimi.it

E-mail: andrea.matta@mecc.polimi.it

E-mail: tullio.tolio@mecc.polimi.it

KEYWORDS

Stochastic optimization, gradient search method, SPSA, simulation-based optimization.

ABSTRACT

Hospital systems are rationalizing and improving their processes in order to provide high quality services at low costs. In particular hospital pharmacies are object of reorganization because of the large wastes in inventories of pharmaceutical and disposable products they produce. Inventory levels in pharmacy warehouses depend on the specific replenishment parameters are used to reorder drugs that are necessary to the hospital. If replenishment parameters are not properly selected, the possibility of incurring in high costs and frequent stock outs increases.

In this context the paper proposes a practical tool for optimizing drug management of pharmacy warehouse in hospital systems. The proposed tool provides to pharmacists the replenishment parameters related to pharmaceutical products managed by the pharmacy. Replenishment parameters are defined by solving a constrained stochastic optimization problem in which the objective function is the total estimated cost for managing drugs in hospitals: ordering costs, inventories costs and costs incurred for backlogging of drugs.

The paper proposes a modified version of simultaneous perturbation stochastic approximation (SPSA) methodology to optimize drug replenishment parameters in hospital pharmacies. Numerical results reported in the paper demonstrates the applicability of the proposed technique to real complex cases.

INTRODUCTION

More and more healthcare systems are pushed to improve their efficiency in order to decrease their cost and to provide high quality services to patients. In this context, healthcare systems reorganize their processes by rationalizing the utilization of resources and using engineering techniques for design and optimization. Among the several organizational units of healthcare systems, pharmacies are considered in this paper because of their role in the total hospitals costs. In fact, inventory costs are around 30 percent of the total costs in the supply chain of pharmaceutical products and the savings that could be obtained by completely reorganizing

the inbound management of pharmaceutical products in hospital are expected to be around 12% (EHCR 1996). However, drug logistics management in hospitals is a critical task since pharmacists have few technical skills (they are physicians), little time to dedicate to logistic issues and often they do not have any technique or tool that supports them in this process. As a consequence, in order to guarantee high quality service and to preserve the patient safety, large inventories characterize often pharmacy warehouses in hospital systems. In particular, pharmacists have to define for every drug the logistic parameters for reordering products when their inventory level is critically reducing. The definition of logistic parameters in hospital pharmacies is a crucial problem that is addressed in this paper.

The paper presents a simulation-based optimization tool for aiding pharmacists to set the drug replenishment parameters in hospital systems. In particular, the proposed tool provides to pharmacists the drug replenishment parameters by solving a stochastic optimization problem in which the objective function is the total estimated operative cost associated to drug management in hospitals. The problem is constrained by a maximum space available in warehouse that limit inventories levels and a maximum stock-out level allowed by pharmacists. Objective function and constraints are functions of drug logistic parameters; however, these functions are not explicit and easy quantifiable with analytical methods. Simultaneous perturbation stochastic approximation (SPSA) is used for finding optimal values of decision variables by conducting several simulation experiments for reproducing the system behavior with stochastic and independent perturbed replenishment parameters.

This paper is the natural evolution of the study in (Albanese et al. 2002) in which the same authors propose a finite difference stochastic approximation (FDSA) method to solve the problem here faced. Indeed, FDSA has demonstrated to perform well and to calculate optimal replenishment parameters in real cases. However, when the number of drugs in the problem increases, the FDSA method is not able to provide solutions in acceptable times. Therefore, the paper presents a modified version of SPSA that use an approximate estimate of gradient with less computational time than FDSA.

The paper is structured as follows. A description of drug management in hospital systems is described in the second

section. The proposed method is described in the third section and conclusions are drawn at the end of the paper.

DRUG REPLENISHMENT IN HOSPITAL SYSTEMS

Description of the system

Pharmaceutical supply chain provides drugs from manufacturers to hospitals through distributors that store products in their warehouses before delivering them to healthcare systems. Pharmaceutical products arrive packed in pallets at the hospital pharmacy and are stored in the central warehouse of the hospital after their de-packaging, control of integrity and coherence with the associated delivery note. When the quantity level of a certain drug is lower than a threshold value, pharmacy emits an order to drug provider specifying number of doses and code of the requested drug. Orders are generally emitted by pharmacy only once a week in order to simplify the replenishment process management. If quantity level in pharmacy warehouse is greater than or equal to the threshold value, no action is taken.

Daily the wards in hospital request a list of drugs to be administrated to patients in the following days. If drugs requested by wards are not available in pharmacy we say there is a stock-out, that is the pharmacy has not in its warehouse enough doses of drugs requested. Stock-outs in hospitals are very critical since unavailability of pharmaceutical products can have dramatic consequences on the safety of patients. When a stock-out is generated, the standard procedure foresees to buy the same type of drug to detailers, or to search the same product in another ward of the hospital, or to substitute the drug with another equivalent product. Depending on the way of solving the problem, the stock-out will cause a different additional cost to the structure. However if there are enough doses of drugs in the pharmacy warehouse, products are delivered, in the following morning, to wards that requested them.

In summary, it is possible to distinguish two different flows related to the management of pharmaceutical products in hospital systems: the physical flow of materials from manufacturers/distributors to wards through the pharmacy, and the information flow of drug requests from the wards to the pharmacy and then to distributors/manufacturers. The control of this flow is that typical of pull systems where orders are emitted when inventory levels are lower than a pre-defined threshold value. Threshold values are different depending on the type of drug, provider's reliability and costs incurred for ordering the drug, for storing it in pharmacy warehouse and for not having the drug when it is requested in the hospital (i.e. stock-out costs).

Pharmaceutical products can be divided into two different classes. The drugs that are very important for the patient safety belong to the first category; for these drugs stock-out should be avoided since the lack of a drug dose can cause the death of a patient. All the other drugs are not so critical and stock-out leads only to the additional costs previously described.

Problem formulation

The analyzed system is composed of pharmacy and wards where drugs are stored in pharmacy warehouse and ward

warehouse respectively. The inventory level of drugs in the pharmacy warehouse increases when an order arrives in the hospital from the distributor or the manufacturer, and it decreases when drug doses are picked and delivered to wards. At the same way, the inventory level of drugs in the ward warehouse increases when drugs are delivered from the pharmacy and it decreases when they are administrated to patients. We define θ^{ROL}_i and θ^{ROQ}_i to be respectively the re-order level and the re-order quantity of the drug type i , with $i=1,2,\dots,p$. When the inventory level of the drug type i is lower than its re-order level θ^{ROL}_i , the pharmacy emits an order of θ^{ROQ}_i doses of drug. Re-order levels and re-order quantity of each drug are the decision variables that have to be defined in order to minimize the following cost objective function:

$$\min L(\theta) = \sum_{i=1}^p c_i^{Ord} + c_i^{Inv} + c_i^{SO} \quad (1)$$

where c_i^{Ord} , c_i^{Inv} and c_i^{SO} are respectively the total costs of ordering, inventories and stock-out related to drug type i ; $L(\theta)$ indicates that total costs are function of decision variables θ . Total ordering costs incurred in a certain period are the product between the number of orders emitted in the period and the cost of orders. As the order quantity increases (with quantity of total ordered drugs assume constant), the total ordering costs decrease because of the reduction of the number of orders; at the same way the total ordering costs increase as the order quantity decreases. Total inventories costs take into account general costs of the warehouse (building, space, energy, etc.) allocated to different drug types proportionally to the total space occupied in the warehouse. In inventory cost is also considered the fact that a certain amount of capital is immobilized in the warehouse instead of investing it in other opportunities. Stock-out costs are the costs incurred when a drug has to be acquired as soon as its inventory has reached the zero level. These costs are very difficult to estimate since the stock-out problem is solved differently depending on the particular case.

The minimization problem is constrained to:

- **Space constraints:** space in warehouse is limited, particularly when drugs have to be stored in environments at controlled temperature; these environments are often very expensive and therefore their capacity is limited and has to be accurately saturated. Space in warehouse is not dedicated but, on the contrary, the different drug types stored in racks share it. This constraint makes the decision variables of different drug types correlated among them, thus forcing to solve the whole problem since it is not possible to decompose it in p independent problems equal to the number of drug types.
- **Service levels constraints:** the average service level of a drug type, defined as the quantity of drug doses delivered to wards over the total quantity of doses ordered by wards, has to be larger than value s_i^* defined by pharmacists. Large values of average service levels means that ward requests are often satisfied and obviously large inventories are necessary in the warehouse. Service values lower than s_i^* are not allowed

and are considered unfeasible in the minimization problem.

Objective and constraints functions are not explicit and depend on decision variables, which are functions of the quantity of drugs administered to patients; this quantity is a random variable that makes the problem stochastic in the objective function and constraints. Another important randomness source is the order lead time, that is the time between the order emission and its arrival to the pharmacy. In the next Section we describe how simulation technique is used to solve the stated problem.

SOLUTION METHODOLOGY

Simulation is used to estimate cost objective function, service levels and space occupied by drug inventories in pharmacy. Simulated time depends on the decision-planning horizon, usually equal to one year in real hospitals and susceptible to relevant reductions if automated tools are used. Analytical methods cannot be used in this case to estimate the system performance because of the strict assumptions they need; for instance, the classical EOQ (Economic Order Quantity) model requires that drug orders and order lead times are normally distributed. Effectively drug requests are sums of the different requests coming from the wards of the hospital; if the number of wards is large, then normality assumption can be correct. However, lead times are strong asymmetric; indeed drugs providers often deliver late, but never before the established date and therefore normality assumption does not hold. From analysis developed on real cases, we can state that normality assumption is often violated in practice.

In the section we propose a modified version of the well-known simulation perturbation stochastic approximation first introduced by Spall in (Spall 1992) to solve large stochastic optimization problems. SPSA is an iterative method that uses an approximate estimate of the gradient $g(\theta) = \delta L / \delta \theta$ to calculate decision variables in the following way:

$$\hat{g}_{k+1} = \hat{g}_k - a_k \cdot \hat{g}_k \quad (2)$$

where \hat{g}_k is the estimate of gradient at k -th iteration and a_k is a positive constant decreasing in the number of iterations k . To estimate the gradient value, decision variables are randomly and simultaneously perturbed according to assumptions defined in (Spall 1992) and two new measurements of objective functions are carried out near the last solution. Then an estimate of the gradient is obtained with the following formula:

$$\hat{g}_{ki}(\hat{g}_{ki}) = \frac{y(\hat{g}_k + c_k \Delta_k) - y(\hat{g}_k - c_k \Delta_k)}{2c_k \Delta_{ki}} \quad (3)$$

where c_k is a positive constant decreasing in k and Δ_k is the p -dimensional random perturbation vector. Notice that the quantities $c_k \Delta_k$ represent the increment randomly obtained for estimating the gradient of objective function. The Δ_k vector has p random mean-zero components, each one being bernoullian distributed and assuming ± 1 values. A set of q independent simulation replications is executed for calculating the function value in perturbed points.

SPSA has been widely used to solve several stochastic optimization problems. Spall provides practical indications for tuning parameters a_k and c_k parameters dependently on the particular considered problem (Spall 1998). However, SPSA applied to the described problem does not provide satisfactory results in terms of convergence speed. This is mainly due to the fact that all variables are treated indistinctly at the same way. Indeed, variables are modified of the same quantity without taking into account the possibility that a variable can assume values that are of some order larger than other variables. If variables have very different sizes, the algorithm has to be tuned on the smallest variable thus increasing the number of total iterations. Similar arguments can be done for gradient estimation and the choice of coefficient c_k . This problem persists even if a_k and c_k are percentage values of decision variables at the last iteration due to the structure of equations (2) and (3).

The paper proposes a modified version of SPSA technique that considers variables with their specific magnitude. In such a way, variables are changed relatively on their values in the last iteration. The method starts from an initial feasible solution θ , a vector with p components. Then independent simulation replications are run to estimate an approximate gradient value of the objective function $L(\theta)$ so that the decision variables are updated with the following recursive formula:

$$\hat{g}_{k+1} = \hat{g}_k - a_k \cdot \text{sign}(\hat{g}_k) \quad (4)$$

where a_k is a vector with p components, each one calculated as follows:

$$a_{ki} = \hat{g}_{ki} \cdot \frac{A_i}{k^\alpha} \quad (5)$$

where A_i are positive constants lower than 1, with $i=1, \dots, p$. Notice that only the sign of gradient is considered to update decision variables since the variable increment (or decrement) is already taken into account in coefficient a_k . In such a way the increment (or decrement) step is in a percentage form so that the algorithm is rough at the beginning but more refined in the last iterations.

In order to estimate the gradient of the objective function, decision variables are randomly and simultaneously perturbed according to assumptions defined in (Spall 1992). Two new measurements of objective functions are carried out near the last solution by means of independent simulations. Then an estimate of the gradient is obtained with equation (3) in which c_k is a vector with p components, each one calculated as follows:

$$c_{ki} = \hat{g}_{ki} \cdot \frac{C_i}{k^\beta} \quad (6)$$

where C_i are positive constants lower than 1, with $i=1, \dots, p$.

If the new solution is not feasible, decision variables have to be modified so that feasible area is again reached. Ad hoc rules have been established to modify decision variables depending on the fact that unfeasibility is caused by the limited space in warehouse or by the minimum service level. The logic adopted by the algorithm to respect the service level is that every time, during k -th iteration that constraints are non respected, associates to variables responsible of

unfeasibility, the values of the $(k-1)$ -th iteration. It's like to associate to the perturbation vector components equal to zero.

The respect of the volume constraint is obtained reducing θ^{ROQ}_i for every drug in proportional manner. Naturally the reduction of θ^{ROQ}_i is subordinated to the respect of the service level.

The algorithm exits when the method converges to a local minimum (absolute minimum is not assured), or when a specified maximum number of iterations is reached or when a satisfactory reduction of the objective function is reached. The value q , i.e. the number of independent replications at each iteration, is defined on the basis of the precision value the user wants on the optimal solution; the classical two-steps method (Law and Kelton, 2000) can be used to calculate q .

NUMERICAL RESULTS

Numerical results are based on a real case in which 10 drugs adopted in an Italian hospital pharmacy have been taken in consideration. For each drug, the following parameters have been collected: storage level at initial period, θ^{ROL}_i and θ^{ROQ}_i , desired service level, historical data on drug consumption in wards, drug management cost, order cost and the distributor lead times. The considered planning horizon in simulation is equal to 3 months, that is replenishment parameters are defined four times per year. Maximum stock-out allowed by pharmacists for each drug is equal to 5%. Volume constraint is not stringent in this particular case and it is relaxed. Notice that decision variables are still correlated, even if at low levels, by the orders emitted to the same producers that group different types of drugs.

The proposed method has been applied to find optimal vales of re-order levels and re-order quantities. In particular the method has been applied by using the following parameters:

$$A_i = 0.1; C_i = 0.1; \alpha = \beta = 0 \quad (7)$$

for each $i=1, \dots, p$.

Table 1 shows replenishment parameters of considered drugs used in the analyzed hospital system. The total expected cost incurred by using hospital parameters is equal to 14.93 Euro/day ± 0.01 at 95% confidence level. Notice that re-order levels and re-order quantities are most of times equal. The reason is that pharmacists have no appropriate tool to correctly select these parameters and therefore the easiest, and also faster, way is to simplify the problem assuming equality between re-order quantities and levels.

In last columns of Table 1, the optimal parameters defined by the proposed method are reported. Replenishment parameters identified by modified SPSA are quite always lower than hospital parameters allowing a significant cost reduction. Exception is for drug type 3; for this drug optimal parameters are larger than hospital values. The reason is that if stock-out constraint has to be respected, level and quantity have to be increased. In fact, a confirm that replenishment parameters used in practice were wrong have been given by the hospital. The total cost using optimal parameters is expected to be equal to 11.3 Euro/day ± 0.05 at 95% confidence level. Optimal value has been obtained after 23

iterations of the algorithm (see Figure 1), each one requiring three function evaluations executed by running $q=20$ independent simulation runs.

Table 1 Real case: hospital and optimal parameters.

Drug Type i	Hospital parameters		Proposed parameters	
	θ^{ROL}_i	θ^{ROQ}_i	θ^{ROL}_i	θ^{ROQ}_i
Drug 1	300	300	161	364
Drug 2	1600	2000	959	1614
Drug 3	300	300	332	360
Drug 4	1200	1200	434	1196
Drug 5	300	600	296	726
Drug 6	200	200	82	161
Drug 7	600	600	301	541
Drug 8	200	200	197	108
Drug 9	2000	2000	1612	1612
Drug 10	500	500	494	328

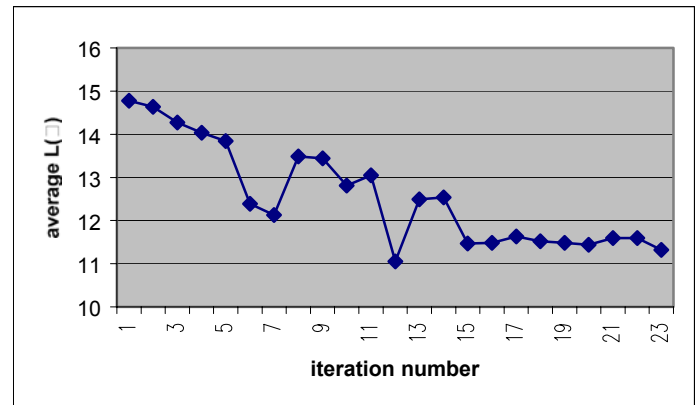


Figure 1. Real case: convergence of the proposed algorithm with 20 variables.

The modified version of SPSA has been applied also to the optimization of a pharmacy with 50 drugs. In this case drug data have been randomly chosen. A practical optimization has been reached in 9 iterations with a cost reduction about 27% moving from an initial value of 194.7 Euro/day ± 2.3 to the minimum value of 165.9 Euro/day ± 5.2 at 95% confidence level. Each function evaluation requires 20 independent simulation runs and coefficients as in (7), except for $A_i=0.2$, are used also in this case.

This problem gives an idea of the large advantage obtained by the use of SPSA in comparison with other methods like FDSA. Indeed, to select replenishment parameters for 50 different drugs means to solve a problem with 100 decision variables. Assuming that also FDSA is applied by launching 20 replications at each function evaluation, at least 2000 simulation replications are necessary at each step; this value can be lower if the number of independent runs is decreased when evaluating the gradient of the function. Therefore,

computational effort necessary to apply FDSA seems to be unpractical in real cases where the number of drugs can arrive to 200. FDSA method converges faster than SPSA in the number of iterations, however each iteration in FDSA requires large efforts. On the other hand, the number of independent simulation run in SPSA for estimating gradient of the objective function does not depend on the number of variables. Indeed, in the case with 100 variables, SPSA requires 60 simulation runs for gradient estimation versus FDSA requiring 2000 runs. This leads to a better gradient estimation for FDSA and, as a consequence, better selection of the next iteration. However, a large experimentation is needed to FDSA iteration and particularly the computational time at each step is proportionally increasing in the number of variable. This case demonstrates the superiority of SPSA respect to FDSA in real hospital pharmacies.

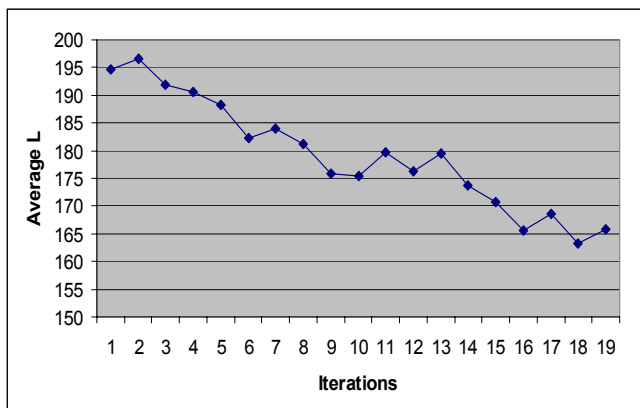


Figure 2. Test case: convergence of the proposed algorithm with 100 variables.

Figure 2 shows the convergence of the proposed algorithm in the presented complex test case. The objective function decreases as the number of algorithm iterations increases. After a reasonable number of iterations the value of the objective function is reduced on average by 15%.

CONCLUSIONS

The paper proposes a modified version of simultaneous perturbation stochastic approximation methodology for finding optimal logistic parameters in the drug management of hospital pharmacies. The proposed methodology fully exploits advantages of SPSA since it is able to solve large practical problems with a lot of variables.

Future efforts will be dedicated to compare the proposed method to other simulation-based optimization methods, such as Response Surface Methodology and likelihood ratio, and to consider other solution points evaluations for better estimating the gradient before to move to the next iteration. Indeed, a drawback of the method is that the current solution is changed after each perturbation even if there is no improvement in the objective function. An interesting idea, which will be investigated in the next future, would be to randomly generate several simultaneous perturbations before moving to the next iteration. In such a way, thanks to a better estimation of the gradient, the probability of going in wrong directions could considerable decrease.

ACKNOWLEDGEMENTS

The research has been funded by the IST project DRIVE "Drug In Virtual Enterprise" IST-1999-12040 funded by the European Community under the V Framework Program.

REFERENCES

- Albanese, M., Matta, A. and Tolio T., 2002. "Defining logistics parameters in the drug replenishment of hospital systems by means of simulation", Italian Society for Computer Simulation Conference, Brindisi, December.
- Andradottir, S., 1998. "Simulation Optimization" in *Handbook of Simulation*, edited by Banks J., Wiley Interscience.
- Efficient Healthcare Consumer Response, Healthcare, 1996. Industry on Efficient Consumer Response.
- Law, A.M., and Kelton, W.D., 2000. Simulation modeling and analysis, McGraw Hill.
- Matta, A., Tolio, T., Albanese, M. and Coppola, P.P., 2001. "Building An Object Oriented Simulation Model By Means Of Arena Templates", *Proceedings of the 2001 ISCS Conference*.
- Spall, J. 1992. "Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation." In *IEEE Transactions on Automatic Control* 37, No.3 (Mar), 332-341.
- Spall, J. 1998. "Implementation of the Simultaneous Perturbation Algorithm for Stochastic Optimization", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, pp. 817-823.

COMPLEX SYSTEMS SIMULATION

COMPLEX SELECTING CRITERIA MODELING WITH COLORED PETRI NETS

Enrique Arjona and Graciela Bueno
Especialidad de Estadística
Colegio de Postgraduados

Km 35.5 Carr Mexico-Texcoco, Montecillo, Edo. de Mex. 56230, Mexico
E-mail: arjona@colpos.colpos.mx, gbueno@colpos.colpos.mx

KEYWORDS

Colored Petri nets, discrete-event simulation, mathematical modeling

ABSTRACT

Colored Petri Nets (CPNs) have been widely used as a mathematical modeling tool of dynamical systems, both for control and simulation. Many processes have been published in the CPN literature but none of them includes complex selecting criteria. In this paper, to broaden the use of CPNs in discrete-event simulation, we present two CPN modules that model complex selecting criteria. The first module selects a constant number of entities from a set of places. The second module selects a variable number of entities from a set of places. The modules were designed to be useful in many modeling situations. Selection of entities is from an unrestricted number of different places, the entities to be selected can be of different types and a set of additional entities might be required in the selection process. Selection processes may be repeated without reinitialization of the modules and the modules can be combined to model more complex selection processes. The paper includes an example of the use of the modules in a real-life simulation model.

1. INTRODUCTION

CPNs have been widely used as a mathematical modeling tool of dynamical systems, both for control and simulation [Gu and Bahri 2002, Salimifard and Wright 2001, Jensen 1997]. Many specific processes have been modeled with CPNs but ordering and selecting criteria, which are very common in discrete-event simulation, have been scarcely explored in the CPN literature. This literature includes only a few models of ordering and none of complex selecting criteria. Martinez et al. [1986] developed a CPN module for a FIFO ordering criterion. Bourcerie and Morel [1997] developed a structure based upon CPNs and FIFO queues to facilitate the modeling of sequential processes. Fanchon [1999] extended CPNs to include FIFO nets. FIFO nets are Petri nets in which all places behave like FIFO queues.

Complex selecting criteria arise when a constant number of entities must be taken from a set of alternative places or when a variable number of entities are required to perform an activity. For example, when a company wants to simulate a layoff policy, when a customer order has to be filled with items taken from several warehouses, when an inspection line can have a variable number of inspectors, or when a vehicle requires a minimum load to start a trip.

In this paper, to broaden the use of CPNs in discrete-event simulation, we present two CPN modules that model complex selecting criteria. Each selecting criterion is modeled by means of a CPN module that simulates the execution of the criterion. The first module selects a constant number of entities from a set of places. The second module selects a variable number of entities from a set of places. Both CPN modules are very general and independent, each defined with its own set of local components, so they can be used to refine existing CPN models. The paper includes an example of the use of the modules in a real-life simulation model [Arjona *et al.* 2001].

The remainder of the paper is organized as follows: Section 2 defines CPNs. Section 3 presents the two CPN modules. Section 4 presents the real life application example. Section 5 gives some conclusions.

2. COLORED PETRI NETS

To make this paper self contained, in this section we include the CPN definition and the CPN terminology that will be used in the following sections. A CPN is a directed bipartite graph $\{P, T, C, I, O, M_0\}$ where:

$P = \{p_1, \dots, p_n\}$ is a set of nodes denominated *places*.

$T = \{t_1, \dots, t_m\}$ is a set of nodes denominated *transitions*.

P and T are disjoint sets.

$C = C(P) \cup C(T)$ is a class of *color sets* associated to the places and transitions such that $\#C(P) = n$ and $\#C(T) = m$ ($\#$ stands for cardinality).

$I: C(P) \times C(T) \rightarrow N \cup \{0\}$ is an *input function* that defines a set of directed arcs from P to T (N is the set of the natural numbers).

$O: C(T) \times C(P) \rightarrow N \cup \{0\}$ is an *output function* that defines a set of directed arcs from T to P .

$M_0: C(P) \rightarrow N \cup \{0\}$ is an initial marking function.

CPNs are mathematical models used to represent discrete-event systems. *Places* are used to store entities. *Color sets associated to the places* indicate the different types of entities that can be stored in the places. Entities are represented by color instances (copies) of the colors associated to the places and are referred as *colors* or *tokens*. Tokens stored in the places, referred as *the marking* of the CPN, define the state of the system. *Transitions* represent sets of related events. Transitions *fire* (are executed) to change the state of the system. *Color sets associated to the transitions* indicate the different ways (events) in which the transitions can fire.

The CPN definition allows only linear input and output functions. These functions are always expressed in mathematical form (as matrices or algebraic expressions) and indicate how many tokens (colors, entities) are stored

or removed from the places when the transitions fire. The CPN definition includes implicitly the passing of time but the definition can be extended to include time explicitly. Temporization can be carried out indistinctly in the places or in the transitions [Murata 1989]. The CPN modules included in this paper do not include time explicitly because the selecting criteria modeled are auxiliary (logical) processes whose execution is virtually instantaneous.

Before execution, a CPN must be initialized (*marked*) with some tokens (colors, entities). CPNs are executed firing one or more of their enabled transitions. A transition is *enabled* (ready to fire), when the tokens required by its input function are available in the appropriate places.

Pictorially, places are depicted by circles, transitions by bars, and functions by directed arcs. Arcs are labeled with symbolic expressions that indicate which colors (tokens, entities) the functions store or remove from the places when the transitions fire (see Fig. 1). Each color is represented with one or more character strings between triangular brackets. When more than one character string is used to represent a color, the color is called a *composite color*.

Despite its conceptual simplicity, CPNs have proved to be a very powerful modeling tool for discrete-event systems, capable of modeling sequences, conflicts, concurrences and synchronization. They provide a formal framework for the design, specification, validation, and verification of discrete systems.

3. CPN MODULES OF SELECTING CRITERIA

In this section, to avoid confusions between the colors (tokens) to be selected and other auxiliary colors used in the modules, the first ones will be referred as entities. Two CPN modules are presented. They model the following selecting criteria:

- 1) Selection of a constant number of entities from a set of places
- 2) Selection of a variable number of entities from a set of places

The modules were designed to be useful in many modeling situations. Selection of entities is from an unrestricted number of different places, the entities to be selected can be of different types and a set of additional entities might be required in the selection process. Selection processes may be repeated without reinitialization of the modules. The modules can be combined to model more complex selection processes. Combination of the modules is straightforward.

3.1 A CPN Module for Selecting a Constant Number of Entities from a Set of Places

Figure 1 depicts a CPN module that simulates the selection of a constant number m of entities from a set of n places. In this module, to increase its modeling power, it is assumed that besides the m entities to be selected, another r additional entities ($r \geq 0$) are also required. This represents a common modeling situation that does not affect the selection process. When $r > 0$, the number of colors used by the module is $6(n+1)+m+2r$.

The execution of the module simulates a manual

selection of the entities. First, it checks if all the required entities are available. The checking is performed by removing and returning the required entities one by one. Each returned entity is counted and marked. The marking avoids considering entities more than once and consists in adding a color to each removed entity. Checking ends when all required entities are marked or when there are no more entities to mark. After checking, if all required entities were available, marked entities are removed from their places and stored without marks in some places outside the module. If not all required entities were available, selection was unsuccessful and marked entities are then removed from their places and returned to them with their marks removed.

The above described process requires to count entities and to fire transitions in sequence. Both situations have to be modeled because in a CPN there is no explicit record of stored entities and all enabled transitions fire simultaneously.

3.1.1 Places and Transitions of the Module

Places PE_i ($i=1,...,n$) have a capacity n_i and will contain the entities to be selected. Each place PE_i ($i=1,...,n$) will contain a different type of entity, one entity at most will be selected from each place, and $m \leq n$. Places PAE_j ($j=1,...,r$) will contain the additional entities. Place PTC will contain auxiliary tokens that will be used for counting the number of marked entities. Place $PSEQ$ will contain auxiliary tokens that will be used for imposing a sequence in the firing of transitions. Transition $TMAE$ will mark additional entities in places PAE_j ($j=1,...,r$). Transitions TME_i ($i=1,...,n$) will mark up to m entities in places PE_i ($i=1,...,n$). Transition TP will test (count) if the number of marked entities in places PE_i ($i=1,...,n$) is m . Transitions $TRMAE$ and $TRME_i$ ($i=1,...,n$) will take marked entities, unmark them, and restore them to their places. Transitions $TOAE$ and TOE_i ($i=1,...,n$) will take marked entities, unmark them, and store them in some places outside the module.

3.1.2 Color Sets Used and Initialization of the Module.

Color sets $E_i = \{ \langle e, i \rangle \}$ ($i=1,...,n$) represent the n entity types allowed in the places. Color sets $NE_i = \{ \langle ne, i \rangle \}$ ($i=1,...,n$) represent absence of entities of the type E_i . Color sets $ME_i = \{ \langle \langle e, i \rangle, m \rangle \}$ ($i=1,...,n$) represent marked entities of the type E_i . Color sets $AE_j = \{ \langle ae, j \rangle \}$ ($j=1,...,r$) represent additional entities. Color sets $MAE_j = \{ \langle \langle ae, j \rangle, m \rangle \}$ ($j=1,...,r$) represent additional marked entities. Color set $TC = \{ \langle tc \rangle \}$ represents the number of marked entities. Color set $S = \{ \langle s \rangle \}$ represents spaces. Color set $SEQ = \{ \langle seq, k \rangle ; k=1,...,3n+3 \}$ will be used to fire sequentially the transitions of the module. Finally, color set $NSE = \{ \langle i \rangle ; i=0,1,...,m \}$ will be used to test if the number of marked entities was m . The color sets associated to the places and transitions are:

$$\begin{aligned} C(PE_i) &= E_i \cup ME_i \cup S ; i=1,...,n \\ C(PAE_j) &= AE_j \cup MAE_j ; j=1,...,r \\ C(PTC) &= TC \cup S \\ C(PSEQ) &= SEQ \\ C(TME_i) &= C(TRME_i) = C(TOE_i) = E_i \cup NE_i ; i=1,...,n \\ C(TMAE) &= C(TRMAE) = C(TOAE) = AE_1 \times \dots \times AE_r \\ C(TP) &= NSE \end{aligned}$$

Each place PE_i ($i=1,...,n$) must be initialized with n_i

tokens (color instances) of E_i or S . Place PTC will be dynamically initialized with m tokens $\langle tc \rangle$ by transition TMAE just before the beginning of the selection process. This dynamic initialization allows the reuse of the module in consecutive selection processes.

3.1.3 Proof of the Validity of the Module

As it was said above, to enhance the modeling power of the module, some additional entities were required in it. The presence of these entities is checked by transition TMAE, so the selection process begins after transition TMAE fires. When the required additional entities are not available, none transition fires and the module is valid by vacuity.

When the required additional entities are available, transition TMAE fires, and the selection process begins. In this case, the proof of the validity of the module has been divided into two theorems. Theorem 1 proves the validity of the module when the selection is not possible. Theorem 2 proves the validity of the module when the selection is possible. Due to space limitations the proof of the theorems and lemmas are not included in this paper.

Lemma 1. In the CPN module given in Figure 1, if transition TMAE fires then the n transitions TME_i ($i=1, \dots, n$) also fire.

Theorem 1. In the CPN module given in Figure 1, if transition TMAE fires and fewer than m places PE_i ($i=1, \dots, n$) contain entities, then (a) one entity of each nonempty place PE_i ($i=1, \dots, n$) will be marked; (b) no entity will be selected; (c) all marked entities will be unmarked.

Theorem 2. In the CPN module given in Figure 1, if transition TMAE fires and m or more places PE_i ($i=1, \dots, n$) contain entities, then (a) one entity of each of the first m nonempty places PE_i ($i=1, \dots, n$) will be marked; (b) m entities will be selected.

3.2 A CPN Module for Selecting a Variable Number of Entities from a Set of Places

A CPN module for selecting a variable number of entities from a set of places may be obtained combining two CPN modules. The first CPN module must select a constant number of entities from the set of places. The second CPN module must select zero or more entities from the set of places. For example, for selecting r to s entities, the first CPN module must select r entities and the second CPN module must select from 0 to $s-r$ entities. The first CPN required module was given in the previous section. The second CPN required module will be given in this section. Combination of the two modules is straightforward.

Figure 2 depicts a CPN module for selecting 0 to m entities from a set of n places. In this module (as in the module of Figure 1) it is assumed that besides the entities to be selected, another r additional entities ($r \geq 0$) are also required. The total number of colors used in the module is $3(n+1)+m+r$.

The execution of the module will simulate a manual selection of the entities. Entities are selected one by one from the places and counted until at most m entities are selected. In this module it is neither necessary to mark entities nor to remove marks because not only m , but any number of selected entities less or equal than m will be sufficient for the selection process.

3.2.1 Places and Transitions of the Module

Places PE_i ($i=1, \dots, n$) have a capacity n_i and will contain the entities to be selected. Each place PE_i ($i=1, \dots, n$) will contain a different type of entity, one entity at most will be selected from each place, and $m \leq n$. Places PAE_j ($j=1, \dots, r$) will contain the additional entities. Place PTC will contain auxiliary tokens that will be used for counting the number of selected entities. Place PSEQ will contain auxiliary tokens that will be used for imposing a sequence in the firing of transitions. Transition TR will remove remaining tokens in place PTC after the selection process is finished. Transitions TOAE and TOE_i ($i=1, \dots, n$) will remove the additional and selected entities and will store them in some places outside the module.

3.2.2 Color Sets Used and Initialization of the Module.

Color sets $E_i = \{ \langle e, i \rangle \}$ ($i=1, \dots, n$) represent the n entity types allowed in the places. Color sets $NE_i = \{ \langle ne, i \rangle \}$ ($i=1, \dots, n$) represent absence of entities of the type E_i . Color sets $AE_j = \{ \langle ae, j \rangle \}$ ($j=1, \dots, r$) represent additional entities. Color set $TC = \{ \langle tc \rangle \}$ represents the number of selected entities. Color set $S = \{ \langle s \rangle \}$ represent spaces. Color set $SEQ = \{ \langle seq, k \rangle ; k=1, \dots, n+1 \}$ will be used to fire sequentially the transitions of the module. Finally, color set $NTR = \{ \langle i \rangle ; i=0, 1, \dots, m \}$ will be used to know the number of tokens to be removed from place PTC after the selection process is finished. The color sets associated to the places and transitions are:

$$\begin{aligned} C(PE_i) &= E_i \cup S ; i=1, \dots, n \\ C(PAE_j) &= AE_j ; j=1, \dots, r \\ C(PTC) &= TC \cup S \\ C(PSEQ) &= SEQ \\ C(TOE_i) &= E_i \cup NE_i ; i=1, \dots, n \\ C(TOAE) &= AE_1 x \dots x AE_r \\ C(TR) &= NTR \end{aligned}$$

Each place PE_i ($i=1, \dots, n$) must be initialized with n_i tokens (color instances) of E_i or S . Place PTC will be dynamically initialized with m tokens $\langle tc \rangle$ by transition TOAE just before the beginning of the selection process. This dynamic initialization allows the reuse of the module in consecutive selection processes.

3.2.3 Proof of the Validity of the Module

When the required additional entities are not available, none transition fires and the module is valid by vacuity. When the required additional entities are available, transition TOAE fires, and the selection process begins. In this case, the proof of the validity of the module has been divided into two theorems. Theorem 3 proves the validity of the module when less than m places contain entities. Theorem 4 proves the validity of the module when m or more places contain entities.

Lemma 2. In the CPN module given in Figure 2, if transition TOAE fires then the n transitions TOE_i ($i=1, \dots, n$) also fire.

Theorem 3. In the CPN module given in Figure 2, if transition TOAE fires and fewer than m places PE_i ($i=1, \dots, n$) contain entities, then one entity of each nonempty place PE_i ($i=1, \dots, n$) will be selected.

Theorem 4. In the CPN module given in Figure 2, if transition TOAE fires and m or more places PE_i ($i=1, \dots, n$)

contain entities, then one entity of each of the first m nonempty places PE_i ($i=1,\dots,n$) will be selected.

4. A REAL-LIFE APPLICATION OF THE MODULES

This section gives an example of the use of the modules in a real-life simulation model. The example models part of the transportation system of a sugarcane plantation [Arjona *et al.* 2001]. The transportation system of the plantation is integrated by 127 trucks, 1 ferry, and 7 roads. Each road connects a different section of the plantation to the mill yard. Four of the roads are continuous and the other three merge at a river crossing served by the ferry. The cane is harvested in the plantation and loaded into trucks. Once loaded, the cane is transported to the mill yard using continuous paths or paths interrupted by the river where loaded trucks have to wait for the ferry, cross the river and continue their travel to the mill yard. The ferry crosses the river only if the number of trucks waiting on the two sides equals a total of four or more. Before the ferry crossing, up to five of the trucks waiting on the ferry side board the ferry (the ferry has a capacity of five trucks). Once unloaded in the mill yard, the trucks return to the plantation or go to the truck yard. When returning to the plantation, the trucks that use the paths interrupted by the river have to wait again for the ferry, cross the river, and continue their travel to the plantation.

The process modeled here is the ferry policy to cross the river. It consists of a preliminary selection of four trucks and a final selection of up to five trucks. In the preliminary selection are considered the trucks waiting on both sides of the river. In the final selection only are considered the trucks waiting on the ferry side. Besides the trucks, in the preliminary selection it is necessary to consider as additional entities the ferry and a ferry operator. The preliminary selection will be modeled with the CPN module given in Figure 1 and the final selection will be modeled with the CPN module given in Figure 2. For the sake of brevity we will refer to these modules as modules 1 and 2. Figure 3 depicts a CPN model of the ferry policy. Module 1 is located in the left.

To model this particular process, a few minor changes have been done to the modules. The sequential tokens used in the two modules are consecutive (module 2 is executed after module 1). Transition TOE of module 1 stores the selected entities in their original places (the selection performed by this module is preliminary). The part corresponding to the selection of additional entities has been suppressed in module 2 (module 2 does not require additional entities). In module 2, the transition name TOE was changed to TOE' (the transition name TOE was already used in module 1). Finally, transitions TME_i , $TRME_i$, TOE_i , and TOE' fire more than once.

Module 1 use two places PAE_j , one for the ferry and another for the ferry operators, so $j=1,2$. Also, module 1 uses two places PE_i , one for the trucks waiting on the ferry side and another for the trucks waiting on the opposite side, so $i=1,2$. Module 2 uses only the ferry side (PE_1) for selecting trucks, so transition TOA' does not need an index. Places PSEQ and PTC are shared by the two modules. The color sets and the color sets associated to the places and transitions of the modules do not change, except for the color set $SEQ=\{<seq,k>; k=1,\dots,3n+5\}$ that contains two extra tokens. The

initialization of the places does not change.

5. CONCLUSIONS

The CPN modules presented in this paper model two selection processes that are very common in discrete-event simulation. They will facilitate the use of CPNs to simulationists. Each module has its own set of independent components, so the modules are ready to refine existing CPN models. The modules can be combined, as was done in the real-life example given, to model more complex selection processes. Combination of the modules is straightforward.

The CPN concepts used in the modules follow the original CPN definition, so available mathematical methods for structural and performance analyses of CPNs [Jensen 1997, Murata 1989] can be used in the CPN models refined with the modules.

ACKNOWLEDGEMENTS

This paper was supported in part with a grant from the National Council of Science and Technology of Mexico (CONACYT) under project 31984-B.

REFERENCES

- Arjona E., Bueno G. and Salazar L., "An Activity Simulation Model for the Analysis of the Harvesting and Transportation Systems of a Sugarcane Plantation." *Computers and Electronics in Agriculture*, Vol 32: 247-264, 2001.
- Bourcerie, M. and Morel, J.Y. "Algebraically Structured Coloured Petri Nets to Model Sequential Processes." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 27, No. 4: 681-686, 1997.
- Fanchon J., "Trace channel nets", in: *Lecture Notes in Computer Science*, Vol. 1630: *Applications and Theory of Petri Nets*, edited by S. Donatelli and J. Kleijn, Springer-Verlag, Berlin, 304-323, 1999.
- Gu T. and Bahri P.A., "A Survey of Petri Net Applications in Batch Processes." *Computers in Industry*, Vol. 47 No. 1: 99-111, 2002.
- Jensen, K. *Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use*, Volumes 1 and 3, Springer-Verlag, Berlin, 1997.
- Martinez, J., Alla, H. and Silva, M. "Petri Nets for the Specification of FMSs." In *Modelling and Design of Flexible Manufacturing Systems*, edited by A. Kusiak, Elsevier Science Publishers, B. V. Amsterdam, 389-406, 1986.
- Murata, T. "Petri Nets: Properties, Analysis and Applications." *Proceedings of the IEEE*, Vol. 77, No. 4: 541-580, April 1989.
- Salimifard K. and Wright M., "Petri Net-based Modelling of Workflow Systems: An Overview." *European Journal of Operational Research*, Vol. 134 No. 3 : 664-676, 2001.

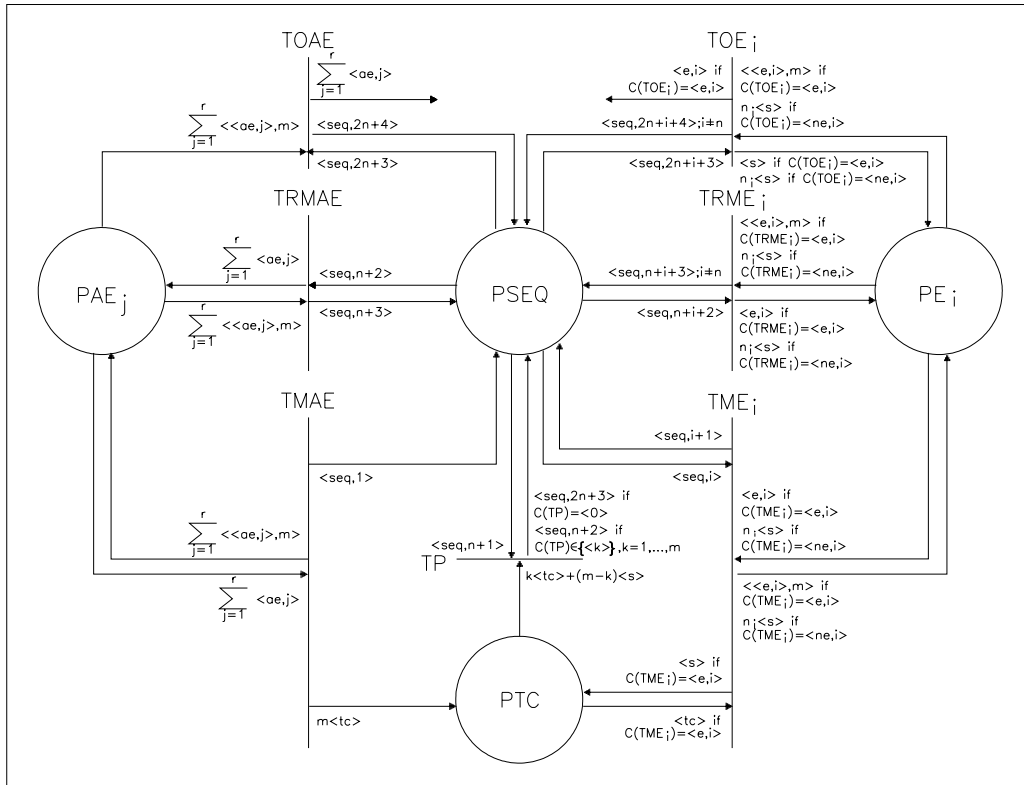


Figure 1. A CPN Module for Selecting m Entities from a Set of n Places

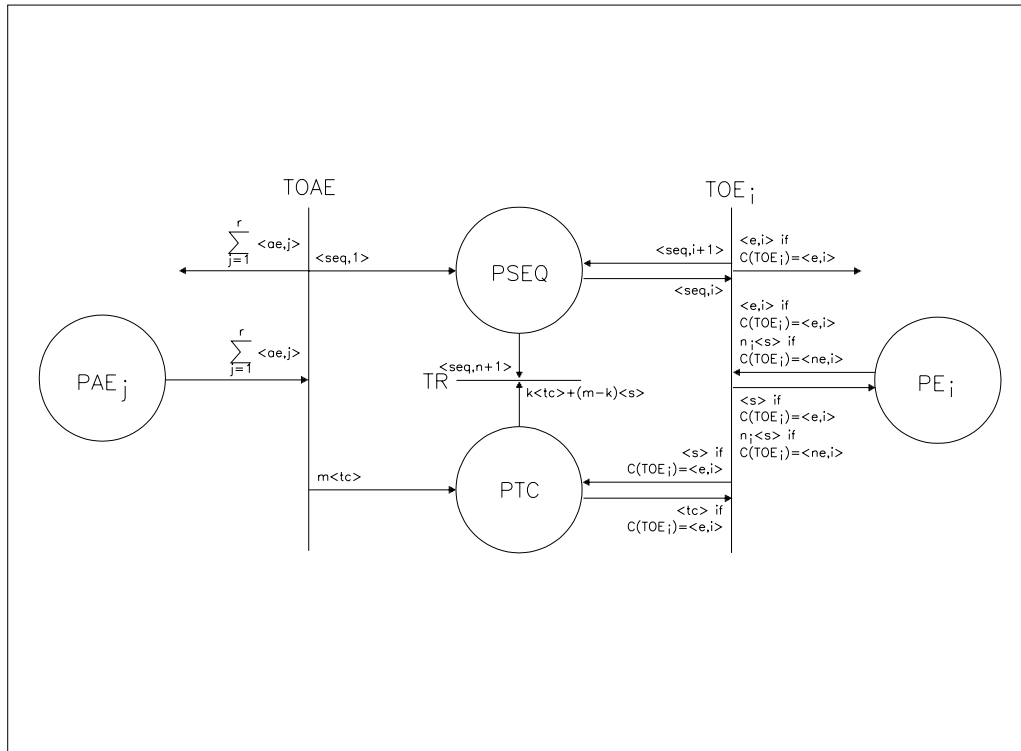


Figure 2. A CPN Module for Selecting 0 to m Entities from a Set of n Places

SIMULATION OF HYBRID CONTROL SYSTEMS USING OPEN PETRI NETS

Stanislovas Bartkevicius, Vidmantas Macerauskas and Kastytis Sarkauskas
Faculty of Electrical Engineering and Control Systems
Kaunas University of Technology,
Studentu 48
LT-3031 Kaunas, Lithuania
E-mail: v.macerauskas@ktu.lt

KEYWORDS

Automatic control, simulation, control systems, system analysis.

ABSTRACT

Modern control systems are complicated and hybrid: continuous processes interact with each other by discrete links. A new simulation tool for analysis of such systems is proposed. Continuous processes are simulated using traditional structural schemes representing differential equations. Discrete links are simulated by a Petri net, which is connected with continuous part of a system model through special, newly introduced, elements – the “processes”. The net may control processes: switch them on or off. The continuous part sends tokens into the net. The specialized software is written to simulate hybrid systems.

INTRODUCTION

The theory of linear systems is well developed and behaviour of such systems can be predicted theoretically. Analysis of non-linear systems is more complicated. Often simulation of a non-linear system is the only way to get reliable results.

A traditional look at control systems as continuous or discrete cannot be applied to modern complicated systems containing both kinds. Large control systems usually are hybrid. They involve continuous, often non-linear, parts as well as discrete ones, or continuous processes are linked by discrete event logic. Systems from energy converters to flexible production lines and complicated ergatic systems are such. The hybrid Petri nets where the marking may be real numbers, are used to simulate systems with a great number of elements. But they cannot be used for simulation of hybrid nets with non-linear, stochastic or discrete phenomena in the analogue part of a model (Le Bail *et al.* 1992).

Investigation of these systems requires new modelling and simulation methods. A model of a hybrid system can be treated as a couple of models – discrete and analogous one, interacting each other. Such a viewpoint is represented in several works (Sahbani and Pascal 2000; Brielman *et al.* 1997; Barton 2002).

The simulation system (Brielman *et al.* 1997) where discrete part of a model works only if a predefined event occurs in an analogue part of the model and interface

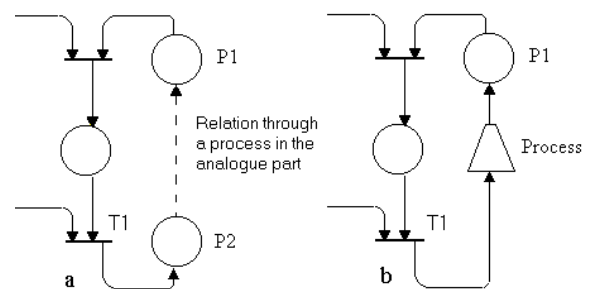
between both parts of a model is realized as special communication tasks is known.

It seems to be more general and promising to apply open Petri nets (Baldan *et al.* 2001) to simulation of hybrid systems. Petri nets are effective means of simulation of discrete systems. A net can represent a control algorithm realized in a numerical controller, for example. Discrete events in the net initiate or fix a new state of the continuous part. The continuous part of a system, looking from the Petri net, can be represented as a set of processes leading from one state to another. These processes are not necessarily known a priori. They may be modelled by a structure simulating continuous differential equations. Some processes may be autonomous. They do not depend on a Petri net, but influence behaviour of the net.

The main purpose of this paper is more detailed description of modelling principles and representation of specialized software for simulation of systems.

TWO PARTS OF A HYBRID MODEL

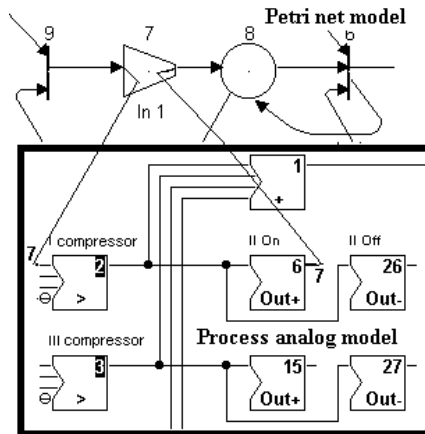
A model of a hybrid system consists of two parts – continuous (analogue) and discrete. The discrete one is a slightly modified open Petri net – in order to make the interface with analogue part more evident a new graphical element representing links with the analogue part and called „process“ (Fig. 1) is included.



Figures 1: Relation of Interface Places through an Analogue Part, a, and its representation by a “Process”, b.

Often a process in the analogue part, initiated by the discrete part of a model, must be checked by the feedback signal, *id est.*, a pair of interface places (Fig. 1, P1 and P2) is related through the analogue part. It seems to be reasonable to show graphically this relation for better understanding of the whole system. It is done by the “process” element (Fig 1, b). A process starts by a signal

from fired transition (Fig 1, T1) and ends by putting a token into a place (Fig 1, P1). Autonomous processes are possible too. They have only output and put tokens into a place. So a Petri net, containing elements („processes“) realizing interface between continuous and discrete parts of a system model, can represent the discrete part.



Figures 2: Invisible Links Between a Petri Net and an Analogue Model

Controllable process is started when the transition connected to its input is fired. As a result of this, the unit step signal appears on an input of a block of an analogue model, which is declared as an input block of the process. This signal may be applied to any block of the continuous part, but the signal from this part to the Petri net must be sent only through blocks of special kind. These blocks (OUT+ and OUT-) produce and put tokens into places connected to outputs of processes.

In order to avoid ambiguous situations an interface between a Petri net and a continuous part must fulfil following requirements:

1. Any process can have only one input (into the wider base of trapezoid) and only one output.
2. Input into the process is allowed only from a transition and output may be connected only to a place.
3. An analogue model can send information into a Petri net only through the blocks OUT+ and OUT-.
4. A controllable process is finished, when a token is sent into a place connected to the output of a process.
5. Attempt to start an active controllable process is an error.

Blocks OUT+ and OUT- are special type of Schmidt trigger sensitive to increasing and decreasing signals.

An analogue part of a hybrid system can be described by a usual for automatic control structural scheme. Only additional blocks for interface with a Petri net are required. It is natural to initiate any process by a standard step signal. This signal can be added to any block of the structure having inputs. The end of a process signal can't be directly sent into a Petri net, but must be converted to a token. This conversion can be done by before mentioned OUT+ and OUT- blocks. There is no need to show links between a net and an analogue structure, when a process has a graphical symbol (element) in the net. These links can be described as parameters of appropriate "processes" (Fig.2).

TIME AND BEHAVIOUR OF A HYBRID MODEL

All processes are under control of a global timer. This timer starts and stops according to status of a Petri net. The rules defining behaviour of a hybrid model assumes that "motion" of tokens in a Petri net needs no time. Delays associated with transitions can be used, if a Petri net simulates not so fast motion. The rules of functioning of a hybrid model are the following:

1. Changes of marking of a net are instant, if delays are not applied to transitions.
2. Activated delays and analogue processes runs (global timer ticks) only if motion of tokens in a Petri net is finished, *id est.*, the Petri part of a model has higher priority.
3. A net is vital until movement of tokens is possible or there are active delays or processes started but not finished.
4. Simulation is finished when value of time exceeds the predeclared value.

The list of rules does not contain usual rules of a Petri net.

Petri net interacts with an analogue model in this way: when a transition connected to the input of a process fires, the unit step signal is set up on an input of a block of the analogue model, which is declared as an input block of the process. This signal remains until the process is finished. The process ends when the block OUT+ or OUT- declared as an output block of the process generates appropriate signal. When a process finishes a token is sent into a place connected to the output of this process.

It is worth noting, that elements of a Petri net may play roles:

1. Position reflects a status of a key, a relay contacts, a switch, a logical element and etc.
2. A transition performs logical AND functions and sends information through the output links about a new changed status (marking) of the control system. If a transition is delayed it plays a role of a delayed relay or other time element.
3. Links represent logical connections of hardware or an algorithm of a control program in a controller.
4. Processes define objects of control or disturbances. A Petri net may influence control object only through processes. On the other hand, an object can change marking of a Petri net only through processes as well.

The introduced simulation scheme does not realize completely the unified framework of hybrid control presented by (Branicky *et al.* 1994), due to properties of used P/T Petri net, but it can be improved if a high-level Petri net used.

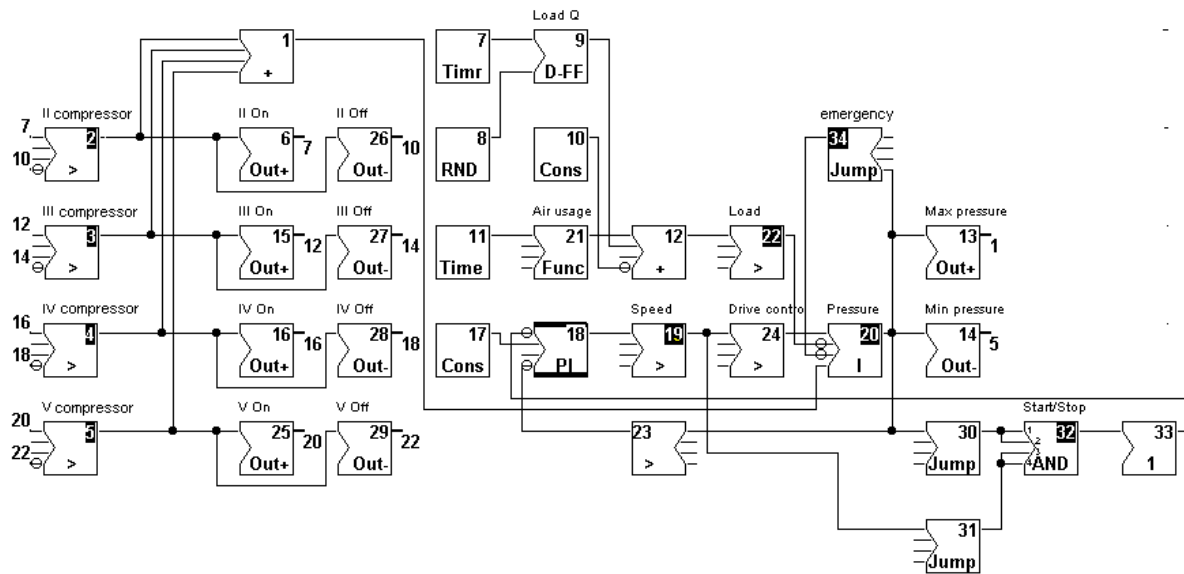
There are some examples of hybrid systems and simulation of them described below.

Model of a compressor station

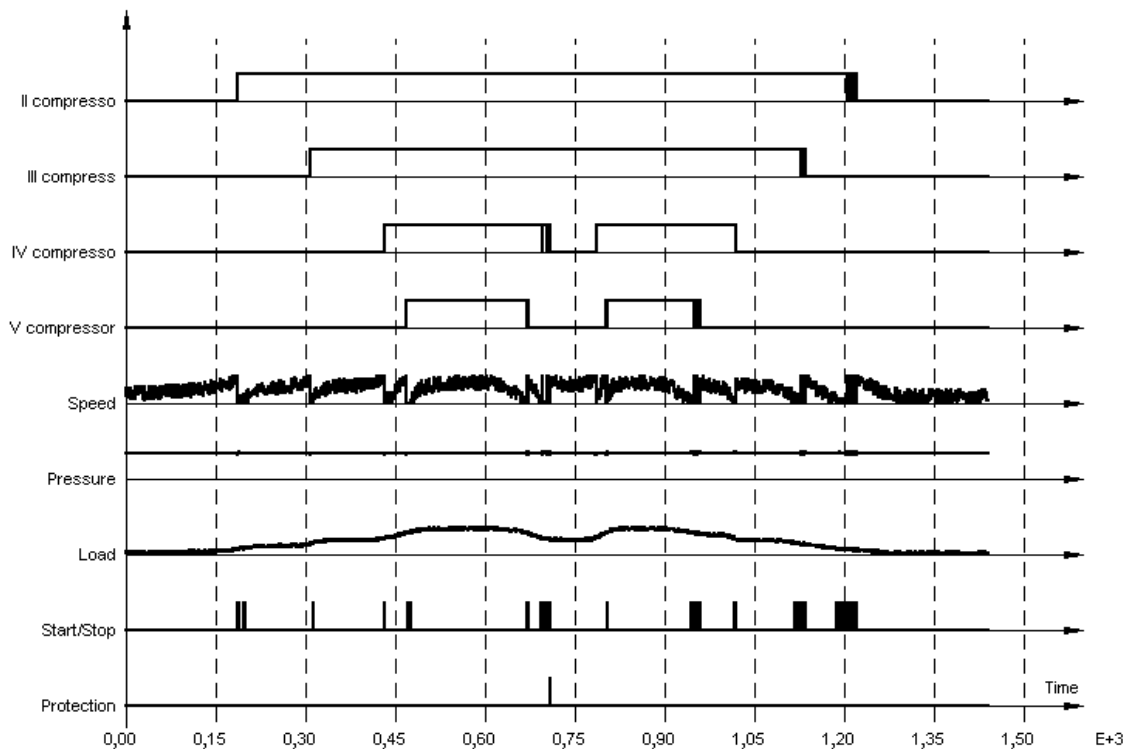
The second example is the model of a compressor station. This model consists of two subsystems: an analogue part, simulating running of the compressor station itself and a Petri net representing a control system or functioning of a dispatch room. The aim of the whole system is to keep air pressure in an air tank in preset limits under random usage

of the air. Usage of the air changes in a very wide range during the day. A Petri net simulates the algorithm of control (start or stop) of additional compressors with respect to changes of pressure in the air tank. The change of

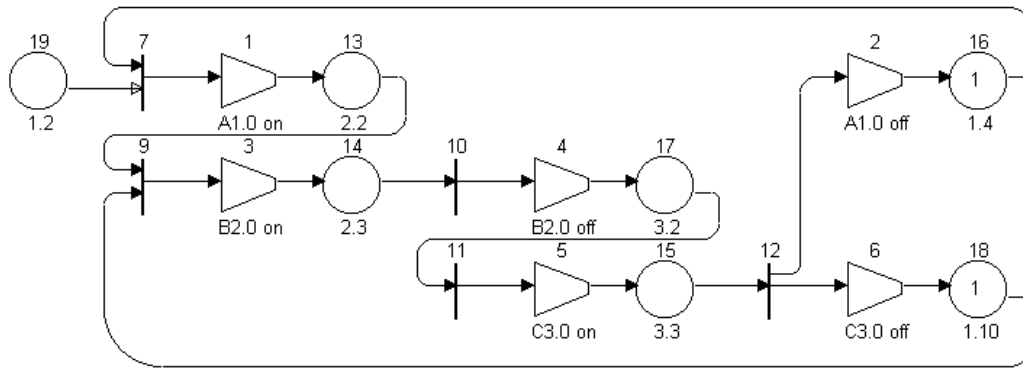
the air pressure is random due to the random usage of the air. Signals from the analogue part setting the pressure out of limits are random too for the same reason.



Figures 3: The Analogue Structure of the Compressor Station. The governed speed compressor is represented by the blocks 19 and 24, 18 – the PI controller of the pressure, 17 – setting of the maximal speed of the first compressor, 23 – the feedback block measuring pressure in the air tank, 20 – integrator simulating the tank, 13 – sensor of the maximal pressure, 14 – the sensor of the minimal pressure. The non-linear block 34 represents the emergency valve. The blocks 30, 31, 32 and 33 represent the start/stop system of the governed speed compressor. The block 21 defines average air usage graphic of the day. The blocks 7, 8 and 9 simulate random dynamic usage of the air. The blocks 12 and 22 represent the total usage of the air. The block 1 sets flows of all switched on additional compressors. The blocks 2 – 5 reflect status of the additional compressors (respectively, from II to V), which are under control of the Petri net. The net is “informed” about the fact of switching on/off of appropriate additional compressor through the blocks 7, 10, 12, 14, 16, 18, 20 and 22. The numbers, next the inputs of the blocks 2 – 5, show the process an appropriate block is associated with.



Figures 4: Time Diagrams of the compressor station

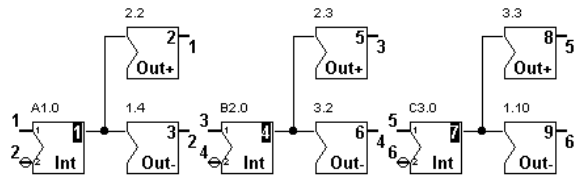


Figures 7: The Petri Net of pneumatic device

2. The cylinder B2.0 may be started, if the cylinder C3.0 is on the initial position (sensor 1.10) and the cylinder A1.0 are on its terminal position (sensor 2.2).
3. The cylinder B2.0 is immediately stopped if it is on the terminal position (sensor 2.3).
4. The cylinder C3.0 starts just if the cylinder B2.0 returns to its initial position (sensor 3.2).
5. When the cylinder C3.0 reaches the terminal position (sensor 3.3), it stops and, as a result, the cylinder A1.0 is stopped.
6. The new cycle can be started only if the cylinder A1.0 is on its initial position.

These conditions are the base to make a Petri net in Fig.7. The analogue part of the model (Fig.8.) simulates only the behaviour of the pneumatic cylinders and sensors. The net simulates synchronization of the device.

A model can be created using afore mentioned rules – a logical part representing as a Petri net and only strictly analogue part realized as a structural scheme.



Figures 8: The Analogue Structure of the cylinders and sensors of the pneumatic device

Another way of modelling is possible too when the entire model can be represented as a structural scheme. It is necessary to use logical elements (triggers, AND elements etc.), only if a structural scheme is applied. It is difficult to see an algorithm in such a structure. The usage of the Petri net seems to be more rational and understandable. Besides, it is very useful on early stages of analysis, when details are not important

CONCLUSION

The models using software CENTAURUS are described in this paper. In the CENTAURUS package a model of the hybrid system has two layers. One of them represents the logical (discrete) part and is defined by a modified Petri net, the other one is the analogue layer, which is described by a conventional structural flowchart. These two layers influence each other through the "process" element in the Petri net.

Any active process sends a step signal to an appropriate block in the analogue layer and this process can be terminated only by a signal from an output block in the same layer. Termination of a process in the analogue layer is defined by occurrence of a pulse on the output signal of a Schmidt trigger. A token is put into a place connected to the output of a "process" element when the process is finished.

Both layers can be simulated separately as well as simultaneously. One global process timer controls all the analogue processes. Both layers of the model are constructed graphically by selecting appropriate elements (blocks) from predefined libraries and by moving them to the necessary places in the Petri net or the flowchart.

REFERENCES

- Baldan P.; A. Corradini; H. Ehrig and R.Heckel, (2001) "Compositional Modelling of Reactive Systems Using Open Nets." Proc. of CONCUR 2001.
- Barton P.I.,(2002) "Modelling, Simulation and Sensitivity Analysis of Hybrid Systems." ACM Transactions on Modeling and Computer Simulation.
- Branicky M.S.; V.S. Borkar and S.K. Mitter, (1994) *Proc. 33rd IEEE Conf. Decision Control, Lake Buena Vista, FL.*
- Briellmann M.; J. Stroop; U. Honekamp and P.Waltermann, (1997) "Simulation of Hybrid Mechatronic Systems" a Case Study, Proceedings of the International Conference on Engineering of Computer Based Systems (ECBS-97), Monterey, California.
- Le Bail J.; R. David and H. Alla, (1992) "Reseaux de Petri hybrids." TSI, vol 11, Nr 5, 95-120.
- Sahbani A. and J.-C. Pascal, (2000) "Simulation of hybrid Systems using stateflow." 14th European Simulation Multiconference (ESM'2000), Ghent (Belgique).

METHODS OF CONCEPTUAL SYNTHESIS OF DYNAMIC MODELS FOR COMPLEX SYSTEMS

Vladimir Putilov, Prof.

Andrey Oleynik, Ph. D.

Andrey Gorokhov, Ph. D.

Kola Science Center of RAS

Institute for Informatics and Mathematical Modelling of Technological processes

Apatity, Russia

E-mail: admin@iimm.kolasc.net.ru

KEYWORDS

Conceptual Model, Dynamic Model, Automatic Mapping, Rules of Deduction.

ABSTRACT

A computerization problem of the design and construction of dynamic models for complex systems has been considered. Authors present formal methods of synthesis of dynamic models on the basis of hierarchical conceptual models. Cases for the task-oriented conceptual model and the object-oriented conceptual model are presented. Rules and steps for automatic mapping the formal conceptual model onto the system dynamics model structure are described.

INTRODUCTION

The System Dynamics Method was developed by J. Forrester in the second part of the XX century. It has recommended itself as an effective approach to study complex systems which are difficult to formalize. Recently this method is widely used in the study of economical and business processes by means of dynamic simulation models (Sternan 2000). However, the automation issue of the design and construction of a dynamic model has been unsolved until now. The view of the problem is the following. A person's understanding about object of an investigation is formulated in terms of basic notions used in the subject field. The structure and composition of these notions are individual for the every subject field. The computer-aided realization of a system dynamics model requires mapping of all collections of characteristics of different realities onto limited collection of the dynamic model's units.

There is the following collection of sets in the dynamic simulation software like Powersim®: {**L**, **F**, **V**, **C**, **R**}, **L** – set of levels (or stocks), **F** – set of flows, **V** – set of auxiliaries, **C** – set of constants, **R** – set of links. Every element of each set has some collection of attributes: the name, the value, etc. So, the dynamic model to be realized in Powersim® is a network with four fixed types of nodes and collection of attributes

The design of a dynamic model became easier with using more comfortable representative form and appropriate notions of a subject field. Using the Casual Loop Diagram is a step towards this direction. But the Casual Loop Diagram doesn't provide the automation of the dynamic model synthesis. Moreover in case of large and complex

systems these Diagrams grow into difficult networked structure.

Hierarchical structures reflect the natural process of the systematic growth of knowledge in many fields of human activity. Authors have developed a Functional-goal technology and a Conceptual design technology based on the use of hierarchical models of the subject field (Putilov et al. 1994). These technologies have developed in various applications from the synthesis of laboratory experiment algorithm to the control of large nature-and-industrial complexes or unique scientific equipment (Putilov 1996; Putilov and Oleynik 2001). The starting point for these technologies is a representation of a goal or of a complex object as a hierarchical system of component subtasks or subobjects. The previous reports (Putilov et al. 1994) in this field have proved that the solution of the main task or the complex object model may be synthesized at the base of easier sub-task solutions or describing the sub-objects according to the decomposition structure. Mentioned technologies are based on the design and analysis of the hierarchical conceptual model of the subject field. Special software was developed to help experts to form the hierarchical representation of the main objects, processes and links in the field of study in accordance with terms of the subject field. The appropriate element of the conceptual model's formal representation is synthesized for every element in the description during the input. Each element is supplemented by a set of attributes. Finally the formal conceptual model (CM) of the subject area is synthesized. It is possible to create the automatic mapping of the formal conceptual model to the dynamic model (Putilov and Gorokhov 2001). Two variants of the dynamic model synthesis based on the formal conceptual model of the subject area will be described further.

THE TASK-ORIENTED CONCEPTUAL MODEL.

In case when a task-oriented approach is dominating the goal task is a root for the hierarchical conceptual model. The CM graph is formed during the decomposition of the goal task into more simple subtasks (Putilov et al. 1994). The terminal nodes of the graph form the set of CM primitives (**P**). This set includes tasks, which are undivided within the frames of given model.

For the resulting set of CM primitives an expert defines the set of actions (**A**) and the set of additional objects (**V_k**), which influence on the actions. The actions we mean the processes, which change the primitives' characteristics.

The formalization of the expert's knowledge in the form of CM provides the basis for the synthesis of a dynamic model. The synthesis of the dynamic model structure is performed by the automatic mapping of CM sets **P**, **A** and **Vk** onto the **L**, **F**, **V**, **C** and **R** sets of the dynamic model. There are three groups of mapping procedures, which are set according to the following rules of deduction:

1. the deduction rules **w1** and **w2** define cover actions set for each of CM primitives;
2. the deduction rules **u1**, **u2** and **u3** define composition and structure (material links) of the dynamic model;
3. the deduction rules **r1**, **r2**, **r3** and **r4** define information links of the dynamic model.

The rule **w1** defines mapping of the CM primitives set onto the set of ordered pairs "primitive-action":

$$\mathbf{w1: P} \rightarrow \mathbf{P \times A}, \quad \forall p_i \in \mathbf{P} \quad \exists a_j \in \mathbf{A}, \quad i = \overline{1, n}, \quad j = \overline{1, k1} : \\ < p_i, a_j > \in \mathbf{P \times A},$$

n is number of CM primitives, $k1$ is number of "negative" or "output" actions, i.e. the actions to decrease numerical value of primitives' characteristics.

The rule **w2** defines mapping of the CM primitives set onto the set of ordered pairs "action - primitive":

$$\mathbf{w2: P} \rightarrow \mathbf{P \times A}, \quad \forall p_i \in \mathbf{P} \quad \exists a_j \in \mathbf{A} : < a_j, p_i > \in \mathbf{A \times P}, \\ i = \overline{1, n}, \quad j = \overline{1, k2},$$

n is number of CM primitives, $k2$ is number of "positive" or "input" actions, i.e. the actions to increase numerical value of primitives' characteristics.

The set **W** ($\mathbf{W} = (\cup \mathbf{P a_i}) \cup (\cup \mathbf{A p_j})$) of pairs "the primitive - the action" ($< p_j, a_i >$) and "the action - the primitive" ($< a_j, p_i >$) is formed by the rules **w1** and **w2**.

The rule **u1** sets the relation between every CM primitive and the elementary object of the dynamic model. It defines the mapping of the CM primitives set (**P**) onto the levels set (**L**) of the system dynamics model:

$$\mathbf{u1: P} \rightarrow \mathbf{L}, \quad \forall p_i \in \mathbf{P}, \quad \mathbf{u1}(p_i) = l_i$$

The set **W** is used in the rule **u2**. The rule **u2** sets the relation between each of the elements of **W** set and the elementary object of the dynamic model. It defines mapping the **W** set onto the flows set (**F**) of the system dynamics model:

$$\mathbf{u2: W} \rightarrow \mathbf{F}, \quad \forall w_j \in \mathbf{W}, \quad \mathbf{u2}(w_j) = f_j, \quad f_j \in \mathbf{F}, \quad j = \overline{1, m}, \quad m \text{ is number of } \mathbf{W} \text{ set elements.}$$

A flow may correspond to one of the three types:

$$f_j = \begin{cases} f_j^+(l_i), & \exists pa_j^i = < p_j, a_i >; \\ f_j^-(l_i), & \exists ap_j^i = < a_i, p_j >; \\ f_j(l_i, l_k), & (\exists ap_j^i = < a_i, p_j >) \wedge (\exists pa_j^k = < p_j, a_k >) \end{cases}$$

$f_j^+(l_i)$ is inflow to level l_i , $f_j^-(l_i)$ is outflow from level l_i , $f_j(l_i, l_k)$ is flow from level l_i to level l_k .

The rule **u3** sets the relation between each element of CM adding objects set (**Vk**) and the auxiliary object of the dynamic model. It defines mapping the **Vk** set onto the auxiliaries set (**V**) of the system dynamic model:

$$\mathbf{u3: Vk} \rightarrow \mathbf{V}, \quad \forall vk_i \in \mathbf{Vk}, \quad \mathbf{u3}(vk_i) = v_i, \quad v_i \in \mathbf{V}, \quad i = \overline{1, s}, \\ s \text{ is number of the } \mathbf{Vk} \text{ set elements.}$$

The rules **r1**, **r2** and **r3** define mapping **W** set onto links set **R** of the dynamic model.

The rule **r1** defines the links depending on the levels:

$$\mathbf{r1: W} \rightarrow \mathbf{R}, \quad \mathbf{W} = (\cup \mathbf{P a_i}) \cup (\cup \mathbf{A p_i}), \quad \mathbf{r1}(w_i) = r(f_i, l_j).$$

The rule **r2** defines the links between flows:

$$\mathbf{r2: W} \rightarrow \mathbf{R}, \quad \mathbf{W} = (\cup \mathbf{P a_i}) \cup (\cup \mathbf{A p_i}), \quad \mathbf{r2}(w_i) = r(f_i, f_j).$$

The rule **r3** defines the links depending on the auxiliaries:

$$\mathbf{r3: W} \rightarrow \mathbf{R}, \quad \mathbf{W} = (\cup \mathbf{A p_i}) \cup (\cup \mathbf{P a_i}), \quad \mathbf{r3}(w_i) = r(f_i, v_j).$$

The rule **r4** sets the relations between the adding objects. It defines mapping the CM adding objects set (**Vk**) onto the links set (**R**) of the system dynamic model:

$$\mathbf{r4: Vk} \rightarrow \mathbf{R}, \quad \mathbf{r4}(vk_i) = r(f_i, v_j).$$

The special matrixes **R1**, **R2**, **R3** and **R4** are generated based on the rules from the third group. Each matrix keeps the mapping schema corresponding to the rule. If i -th element of the CM maps onto j -th element of the system dynamic model the element \mathbf{rx}_{ij} of the matrix equals 1. The element \mathbf{rx}_{ij} equals 0 otherwise.

$$\mathbf{rx}_{ij} = \begin{cases} 1, & \text{if } e_{cmi} \rightarrow e_{dmj}; \\ 0, & \text{if not } (e_{cmi} \rightarrow e_{dmj}). \end{cases}$$

These are the formal rules for mapping of the hierarchical CM primitives onto the network structure of the system dynamic model. Mapping procedure algorithm is cashable. It is afford for computerization of the dynamic model synthesis. Figure 1 represents the synthesis steps for the dynamic model structure based on the task-oriented conceptual model.

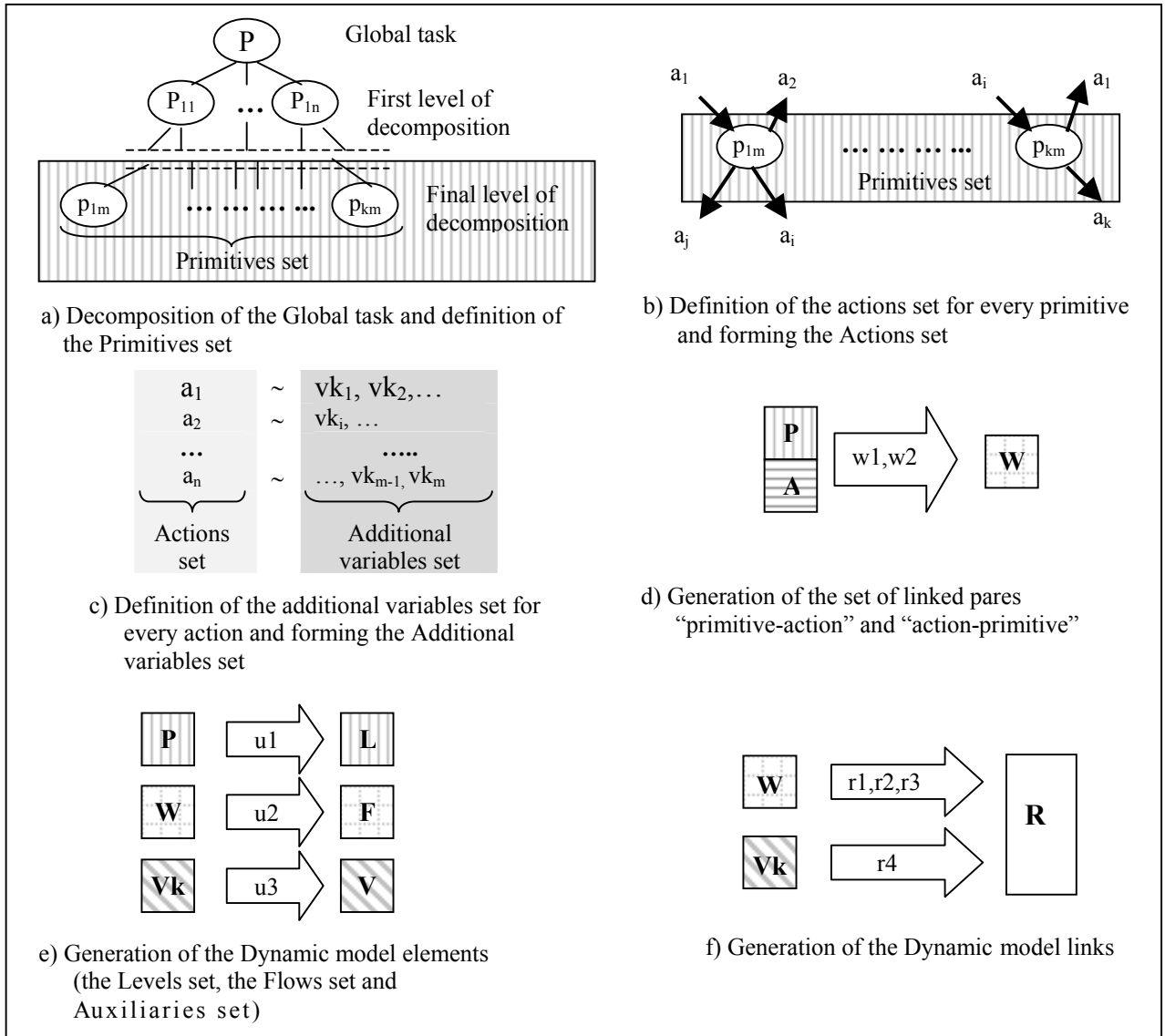
THE OBJECT-ORIENTED CONCEPTUAL MODEL

The structure of the conceptual model, which is based on the object-oriented approach, differs from the task-oriented CM. It is based on two interrelated tree-like graphs (Putilov et al. 1994). One graph represents the hierarchy of the system of objects (or recourses) to be investigated. The second graph represents the structure of actions (or processes) of the system to be investigated. Each action is interpreted as a change of the value of the recourses or value of the recourse characteristics. The input/output relations are linked between the objects and the action graphs. So, the object-oriented CM consists of the following sets collection:

$$\{\mathbf{A}, \mathbf{O}, \mathbf{H}, \mathbf{In}, \mathbf{Out}\},$$

A is the set of actions (or processes) in the system, **O** is the set of objects (or recourses), **H** is hierarchy relations on the set of action ($\mathbf{H_A: A \rightarrow A}$) and on the set of objects ($\mathbf{H_O: O \rightarrow O}$), **In** and **Out** are the sets of "input/output" relations (links) between the actions and the objects. The hierarchy relations define the representation of some hierarchical level object or action as the more simple objects or actions on the next hierarchical level.

The CM in declarative form represents a complex object or process as a system of more simple conceptual models of sub-objects or sub-processes. Therefore CM allows analyzing any CM part as an independent model of a corresponding subject area fragment (subsystem). Formal structure of CM objects or processes is identical on each of hierarchical levels excluding the terminal level. The hierarchy relations are not set (determined) for terminal level because the internal structure of terminal objects and



Figures 1: Steps of a Dynamic Model Synthesis. (Task-oriented Conceptual Model case)

processes are not considered in the frame of the model. Identity of the different levels of formal representation predetermines possibility for integration of existing models into more complex model of higher level and consecutive specification by terminal level decomposition.

Only one level is selected from each CM subtree for system dynamic model design. The limitation is the existence of horizontal links or possibility that links can be identified between the selected levels of various subtrees in treelike CM graph. The obtained cross-section $\{A', O', In', Out'\}$ is a basis for the synthesis of a system dynamics model.

Each of CM actions and CM objects are attributed by the sets of characteristics. Three attributes have high significance for dynamic model synthesis computerization. They are: name, value and evaluation function ($\langle n, val, f \rangle$). The name further will be used for identification. The evaluation function calculates the depending object value on external factors and over objects values ($val_i = f_i(x, \{val_j\}), j \neq i$). In the case when the evaluation function is not set ($f_i = nil$) the starting value is a constant ($val_i = val_{0i} = const$). In the case when the function doesn't

include values of over objects ($\{val_j\} = \emptyset$) that means the object value depends only on external factors ($val_i = f_i(x)$). As in the case of task-oriented CM it is necessary to determine the rules for obtained formal CM cross-section $\{A', O', In', Out'\}$ mapping onto the sets of the system dynamics model $\{L, F, V, C, R\}$.

The flows set of synthesized dynamic model is defined by the actions set included into CM cross-section:

w1: $A' \rightarrow F, \forall a_i \in A', w1(a_i) = f_i$.

Elements of the object (or objects) set may be mapping onto elements of one of the three sets L, V or C depending on the attributes values.

In the case when the evaluated function is a function of input and/or output actions values the object is represented by level of the dynamic model.

w2: $O' \rightarrow L, \forall o_j \in O', w2(o_j) = l_j$ If $[val(o_j) = f_i(x, \{val(a_i)\}) \wedge (a_i \in in(o_j) \vee a_i \in out(o_j))]$.

In the case when the evaluated function is not a function of input and/or output actions values and is a function of over objects values the object is mapping onto auxiliary of the dynamic model

w3: $O' \rightarrow V$, $\forall o_k \in O'$, $w3(o_k) = v_k$ If $[val(o_k) = f_i(x, \{val(o_j)\}), j \neq k]$

In the case when the evaluation function of the object is not set ($f_i = nil$) the object is mapping onto constant:

w4: $O' \rightarrow C$, $\forall o_m \in O'$, $w4(o_m) = c_m$ If $[val(o_m) = const]$.

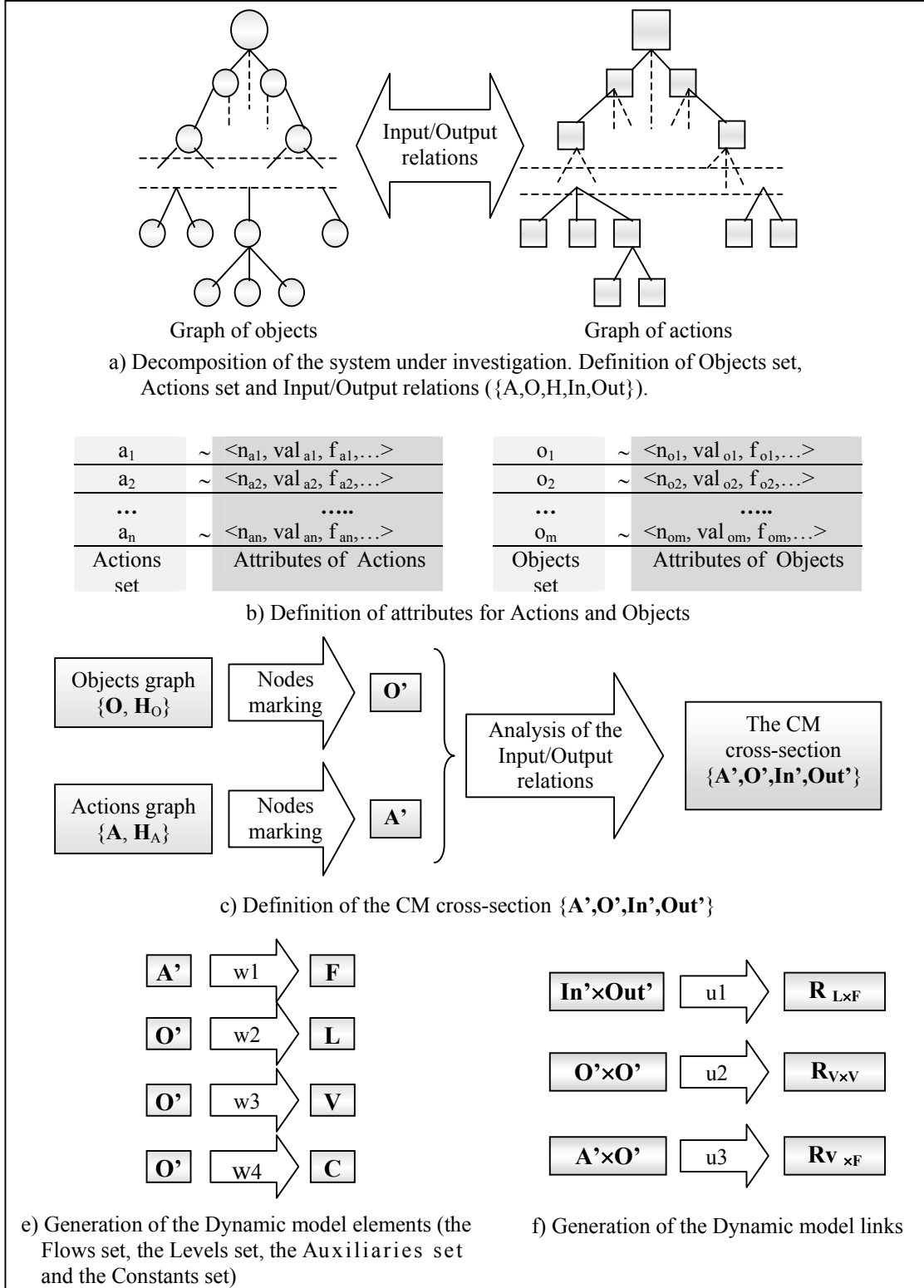
The sets $F = \{f_i\}$, $L = \{l_j\}$, $V = \{v_k\}$ and $C = \{c_m\}$, that are obtained as a result of the represented above mapping, define composition of the system dynamics model.

The next step in generation of a dynamic model is the synthesis of links. The links are synthesized on the basis of the input/output relations and on the basis of analysis of the evaluation functions structure.

The input/output relations define the “inflow-level” and “level-outflow” links:

u1: $In' \times Out' \rightarrow L \times F$.

If evaluation function of object o_i uses value of object o_j the information link r_{ji} is set from auxiliary v_j ($o_j \rightarrow v_j$) to



Figures 2: Steps of a Dynamic Model Synthesis. (Object-oriented Conceptual Model case)

auxiliary v_i . ($o_i \rightarrow v_i$) in the dynamic model:

u2: $O' \times O' \rightarrow R$, $\forall \langle o_j, o_i \rangle, o_j \in O', o_i \in O'$,

u2($\langle o_j, o_i \rangle$) = r_{ji} If [$f_{oi} = f_i(x, \{Val\}_i) \wedge val_{oj} \in \{Val\}_i$]

The links for the dynamic model rate-of-flow variables is defined similarly on the basis of evaluation functions of CM action value analysis:

u3: $A' \times O' \rightarrow R$, $\forall \langle a_j, o_i \rangle, a_j \in A', o_i \in O'$,

u3($\langle a_j, o_i \rangle$) = r_{ji} If [$f_{ai} = f_i(x, \{Val\}_i) \wedge val_{oj} \in \{Val\}_i$].

Figure 2 represents steps in the synthesis of the dynamic model structure based on the object-oriented conceptual model.

CONCLUSIONS

Both task-oriented and object-oriented declarative conceptual models are effective tools for formalization of experts' knowledge on the object of investigation. Hierarchical structure of conceptual model and the possibility to use appropriate notions from the subject field make the modeling process easier. The structure of the formal representation of the conceptual model doesn't depend on the specific subject area and allows broad range of applications of the conceptual design methods. The problem of the design phase of complex system dynamics model computerization is solved by using the developed

rules and algorithms for mapping the formal conceptual model onto the system dynamics model structure.

REFERENCES

- Putilov V.A., V.V. Filchakov and A. Ya. Fridman. 1994 *CASE-technologies for computer-aided experiment*. Apatity, KSC RAS. (in Russian)
- Putilov, V.A. 1996. "Knowledge based models for production planning under unstable economy". In *Proceedings of the European Simulation Multiconference* (Budapest, Hungary, June 2-6), 1018-1022.
- Putilov, V and A. Gorokhov. "Conceptual Projecting of System Dynamics Models (On the example of creating a Dynamics Model of Town development)" In *Proceedings of the Modelling and Simulation 2001, 15-th European Simulation Multiconference* (CTU Prague, Czech Republic, June 6-9), 109-112.
- Putilov, V.A. and A.G. Oleynik. 2001. "Information technologies and the problems of sustainable development of industrial regions in the Russian Arctic" In *Proceedings of the Joint EU-Russia-Canada-US Workshop*. (Brussels, Belgium, Oct. 25-27), 88-97.
- Sterman J. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. McGraw-Hill

VISUAL REPRESENTATION OF ENHANCED SAND PILE MODELS

Marta Pla-Castells
Ignacio García
Rafael J. Martínez
Instituto de Robótica
Universidad de Valencia
P.O. Box 2085
46071 Valencia, Spain

E-mail: [Marta.Pla;Ignacio.Garcia;Rafael.Martinez]@uv.es

KEYWORDS Cellular automaton, simulation, sand pile model.

ABSTRACT

This paper overviews some models based on the Chip Firing Game and propose a modification of the Abelian Sand Pile Model in a lattice in order to implement a realistic simulation of a pile of sand falling and finding an stable configuration. We have modified the update rule in the way that now is based on the comparison among neighbour nodes. We found that the simulations can be suitable for doing them in real time and propose as a future work a better solution to reduce the computational cost of the algorithms.

INTRODUCTION

Nowadays, simulation in real-time is applied in many fields concerning interactive games or in those activities related with instructional designs. In these cases, not only is essential to have dynamic models with good accuracy but also it has to be taken into account the visualization techniques used in order to obtain a good degree of realism.

Usually, when dealing with simple rigid objects, like cars, trains or containers, the main efforts of the designers are addressed to obtain dynamic models with good interaction with the virtual world. This is an easy task and forces can be calculated very well in real-time. By the other hand when complex objects or deformable bodies are being simulated, like the case of hair movement, cloth behaviour, water or fire simulation, modelling tasks are not so easy as before but however, their visualization properties have been deeply studied by many researchers. There are still another kind of elements whose properties and behaviour require

to pay attention at the same time, both to its visualization properties and to its interactions with environment. This is the case, by example, of multiple particle systems subjected to human manipulation, like in the case of a terrain movement machine simulation or in a coal mobil crane simulation. In such systems, accuracy collision detection algorithms, complex models and enhanced visualization techniques have to cope with real time simplifications in order to satisfy the effects of the user actions over the simulation equipment.

When modelling the dynamics of a mountain of grain and its topplings, a good election, at least from the point of view of its theoretical properties and unestabilities, could be to use the sand pile model, since it has been deeply studied for many years (Bak et al., 1988). However, as it will be shown later, these classical algorithms exhibit a poor realism when the evolutions of the sand pile are being displayed in real-time using 3D visualization techniques.

This paper describes a new algorithm, which is a modification of the sand pile model, that represents very well the visual properties of such kind of piles when evolve until stability. In the first section an introduction to the Self-organized Criticality (SOC) is shown. Following, it relations with other models used to represent sandpiles, like the Chip firing game (CFG) and its variants, the Abelian Sand pile model in a lattice (ASPM) and the Sand pile model, are studied, and the results obtained from the computational implementation of ASPM in order to represent a mountain of grain are also shown. In second section we propose some modifications to the original models in order to achieve better accuracy taking into account that we are interested in its visual representation. Finally, third section shows some ideas for future work that will allow reduce the temporal cost of the algorithm at every step.

SELF-ORGANIZED CRITICALITY

The concept of Self-Organized Criticality (SOC) is used to explain a particular behaviour that is observed in many natural systems and processes. These systems are generally characterized by the existence of a power law that rules their dynamics. The term self-organization denotes that in many critical systems tend to appear some structure or pattern at the macroscopical level. Examples of these kind of systems can be a sand pile, the frequency and magnitude of extinctions that take place in every ecosystem, the distribution of star clusters along the outer space, or the expansion process of Internet nodes. SOC related models have also been applied successfully in computational economy and social studies, like the work done by Scott Moss dealing with social simulation (Moss, 2001).

The Sand-Pile Model (SPM) was introduced by Bak, Tang and Wiesenfeld (Bak et al., 1988) in order to represent a self-organized critical system by means of a cellular automaton. This model received its name because of the analogies observed among the dynamical rules of the model and the way sand topples in a real life sandpile. Other studies, like the one done by Brylawski (Brylawski, 1973), use the properties of a one dimensional SPM to study integer partitions. SPM models are also useful to represent the information paths that take place in parallel architectures and distributed computing. In this sense, good performance is obtained when using the gradient properties of SPM to implement a dynamic load balancing algorithm (Subramanian and Scherson, 1994). In such case, the grains of the piles represent data or tasks, and the stacks formed by the grains, the set of available processors.

All the algorithms viewed before adjust the parameters of SPM to resolve a complex problem. In our contribution we want to take benefit of some of the models appeared during the study of this subject not for resolve a mathematical problem, but for testing its suitability to make a realistic visual representation of sand piles.

In next paragraphs we will review some of the different sandpile models appeared in literature during the last years that will be the basis of our algorithm.

Chip Firing Game

The Chip Firing Game (CFG) (Björner and Lovász, 1992) can be defined as a directed graph whose vertexes, v , have two parameters associated, a threshold, δ_v , and a load, $l(v)$, that intuitively represents the number of chips stored in v . A general rule is applied to decide the new load of the vertexes at every step. An SPM is a particular case of CFG.

In the model presented by (Björner and Lovász, 1992) the game evolves with respect to the following rule: if a vertex v contains more than δ_v chips, then δ_v of them

are fired to its neighbours, i.e. the load of the vertex v is decreased by δ_v and the load of each of its neighbours is increased by δ_v/n_v , being n_v the number of neighbours of v . Usually the value elected for δ_v is equal to n_v .

Abelian Sandpile Model

Another particular case of the CFG is the abelian sandpile model, which is just an SPM over a rectangular grid (Dhar, 1990). In this case, the model is an undirected graph that forms a rectangular finite grid. The value of δ_v and n_v is always 4 for every vertex, except for an additional vertex, which receives one link from the vertexes located at the border and two links from the four corner vertexes. This special vertex also has an infinite threshold δ_v that makes it act like a sink; it collects the grains that leave the system and it never gives any of its grains.

All the models derived from the Chip Firing Game exhibit the property that are strongly convergent games (Eriksson, 1993). This property ensures that given an initial configuration, either a game can be played forever or it reaches a unique fixed state (where no firing is possible), called the *final configuration*. Furthermore, this state does not depend on the order in which the vertexes are fired.

Dhar (Dhar, 1990), based on this property, showed that in ASPM, for every initial configuration, some final configurations will be never reached (are forbidden), and that the others, the *allowed ones*, will happen with equal probability and in a finite number of steps.

In (Dorso and Dadamia, 2002) we can find a prediction study of every evolution step in an ASPM that represents an avalanche for a given size and in (Moore and Nilsson, 1999) it is demonstrated that the problem to know which will be the final state of an ASPM, given an initial configuration, is P-Complete.

1-Dimensional Sand Pile Model

Another special case of CFG is the Sand Pile Model (SPM) introduced by (Latapy et al., 2001). The graph, in this case, is an undirected chain, infinite on the right, whose vertex have a unique connection to the next. The threshold δ_v is equal to 2 for every vertex, except for the first one, which equals to infinity.

As in the previous models, most of the works related to this are quite theoretical (Durand-Lose, 1996; Goles and Latapy, 2002; Eric Goles and Phan, 2002; Novelli and Rossin, 2001; Latapy et al., 2001). However the novelty of them is that they introduce a variation in the updates of the sand pile. The general rule to decide whether a node must fire or not is calculated by means of a comparison with the load of the next neighbour, and not using the absolute value of the local node.

This variation will be used in our work to obtain similar 2-dimensional models more suitable for visualization purposes.

Implementation Details of ASPM

When working with ASPM based models, it is natural to have in mind the idea that we are representing a pile of sand, like those that exist in deserts or beaches, that evolve along the time. The question we are going to resolve in this section is whether these models represent with accuracy not the behaviour of such systems but their properties related with their visual appearance. The objective of this task is to achieve a good algorithm that can be used in a simulation environment where sand piles, or by extension particle systems, can be modelled according to the point view of their graphical shape.

For visualization purposes, an ASPM model can be displayed using a mesh whose nodes are loaded with a value, like the graph nodes of the theoretical model. This technique is used to represent the shape of the sandpile, not the total amount of sand grains, and therefore the problem is simplified. Also, we are going to suppose for our tests that adding sand to the pile causes the same effect than raising a zone of the mesh and leaving it to evolve to the stable state.

In order to implement the computational algorithm of an ASPM, we have used the method explained in (Dhar, 1990) which states that the number of allowed configurations is equal to the determinant of an integer matrix Δ that specifies the evolution rules.

Δ is a $N \times N$ matrix where $\Delta_{ii} = n_v$ and $\Delta_{ij} = -1$ for each node j connected to i , so that row i of Δ represents the amount of sand lost by every node when node i fires.

According to the rules of ASPM, the matrix takes the form

$$\Delta_{ij} = -1 + (n_v + 1)\delta_{ij}$$

where δ_{ij} is the Kronecker delta (δ_{ij} is equal to 1 if $i = j$ and 0 elsewhere) and n_v is equal to 4 since represents the number of neighbours considered.

Since ASPM is a strongly convergent game, it is guaranteed that the pile will reach the same final configuration independently of the way the grid nodes are updated. This property is very useful since will allow parallelize the algorithm and therefore implement it using a multithreaded application with increases performance.

Test Results

The computational implementation of this matrix has allowed us perform some tests in order to discover the similarities among the real life and the modelled systems.

All the simulations have been done using a grid of size 30×30 . At the beginning of the simulation process,

the cells in the rectangle $[10, 18] \times [16, 20]$ have been initialized randomly with $l(v) \in [0, 30]$. With this initial conditions the simulation is run until the system reaches the *final configuration*.

Algorithm 1

```

for  $i = 2 : N - 1$ 
  for  $j = 2 : N - 1$ 
    if  $(M(i, j) > \delta_v)$ 
       $M(i, j) = M(i, j) - \delta_v$ ;
      for  $p = 1 : n_v$ 
         $k = \text{neigh}(p, 1)$ ;
         $l = \text{neigh}(p, 2)$ ;
         $M(i + k, j + l) = M(i + k, j + l) + \delta_v / n_v$ ;
      end
    fi
  end
end

```

Algorithm 1 shows a pseudo-code implementation of ASPM. In this case, as we have seen above, we will use $\delta_v = n_v$.

The experiment has been repeated for two different definitions of the neighbouring of a cell. The first one uses the following expression:

$$\text{neigh}_4(i, j) = \{(i + 1, j), (i - 1, j), (i, j + 1), (i, j - 1)\}$$

and therefore $n_v = 4$. This neighbouring is the most commonly used, and some examples can be found in (Dhar, 1990) or in (Dorso and Dadamia, 2002).

The second neighbouring tested has been defined as:

$$\text{neigh}_8(i, j) = \text{neigh}_4(i, j) \cup A$$

where

$$A = \{(i + 1, j + 1), (i + 1, j - 1), (i - 1, j + 1), (i - 1, j - 1)\}$$

In this case, $n_v = 8$.

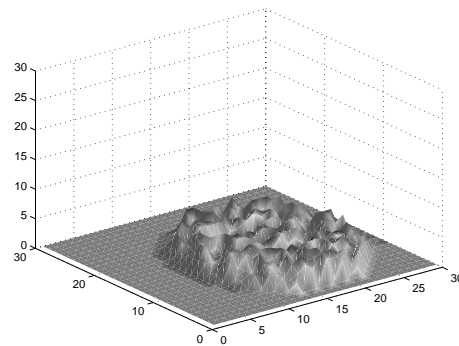


Figure 1: ASPM with neigh_4

The simulations were performed using *MATLAB*TM. The cpu-time measured to implement one step was 0.023

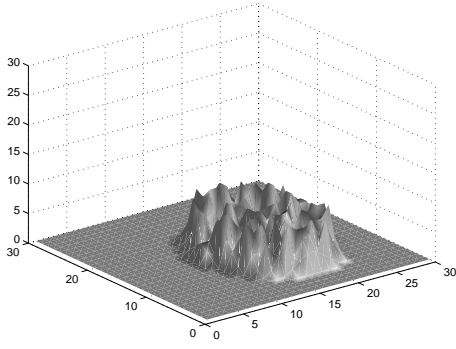


Figure 2: ASPM with $neigh_8$

for the $neigh_4$ case and 0.025 for the $neigh_8$ case. This includes the redrawn of the mesh. It is also worth noting that the number of steps required to achieve the final state is quite significative. In the case of $neigh_4$ was 104 and for $neigh_8$ was 57.

Figures 1 and 2 show the *final configuration* for the simulations. The results show that the ASPM, as it was originally presented, does not behaviour like grain piles or terrain mountain do. The main reason is that the height of any cell is bounded by the threshold of the model, and the system spreads wide until this requirement is satisfied.

For our purpose of simulate sand or grain falling and forming a mountain, this shape of a sand pile is not realistic at all. In the next section we propose some modifications to the original ASPM that will provide more realistic shapes suitable for visualization applications.

SPM FOR VISUAL APPLICATIONS

As can be seen in the previous section, the original ASPM model does not suit very well with the purposes of simulating a realistic sand pile. Most physical grain flow models state that one of the properties of a sand mountain is the angle formed by the slope, which depends on the material properties. In this section some changes in the actualization rules of the models are proposed to cope with this characteristic.

Enhanced Sand Pile Model

We are going to consider a modification of the Abelian Sandpile Model by means of changing the update rule. In the original model an absolute threshold δ_v was used. Whenever the height of a cell overpasses δ_v it fires, causing sand topples.

The new model has to emulate the actual behaviour of sand, characterized by the occurrence of toppings only when the mountain shows high slopes. According to this, every cell will now be compared to its neighbours. If the difference in the height of two adjacent

cells overpasses a certain threshold δ_t , the higher cell fires a number of grains z_+ into the lower one.

This model has two parameters, instead of one, and there exists no relation among the threshold and the number of neighbours, as there was in the ASPM. These parameters will allow control some aspects of the final state of the system, such as the angle of the slope or the frequency and length of toppings. Algorithm 2 shows its implementation details.

Algorithm 2

```

for  $i = 2 : N - 1$ 
  for  $j = 2 : N - 1$ 
    for  $p = 1 : n_v$ 
       $k = neigh(p, 1);$ 
       $l = neigh(p, 2);$ 
       $dif = M(i, j) - M(i + k, j + l);$ 
      if ( $dif > \delta_t$ )
         $M(i, j) = M(i, j) - z_+;$ 
         $M(i + k, j + l) = M(i + k, j + l) + z_+;$ 
      fi
      if ( $dif < -\delta_t$ )
         $M(i, j) = M(i, j) + z_+;$ 
         $M(i + k, j + l) = M(i + k, j + l) - z_+;$ 
      fi
    end
  end
end

```

Some aspects must be considered within the new model. When talking about the ASPM, the concept of strong convergent model was introduced. In this case no guarantee exists that the new model will have even a *final configuration*. This point is not crucial for the objective of this work, though some later studies should be done to clarify it.

Another question, more closed to the subject of this paper, is the decision of how the new rule has to be run. Strictly, the new rule does not determine which of the cells in the neighbouring must be checked first. Two variants have been used in this paper, one with a fixed order of the neighbours and the other taking the neighbours randomly.

Test results

We repeated the simulation made with the ASPM model in order to compare the results of both methods for different values of δ_t and z_+ .

We choosed values for δ_t between 2 and 8 and for z_+ between 1 and 8. The results of the simulation showed that if the difference among δ_t and z_+ was small, the visual appearance was better than if this difference was high. On the other hand as higher δ_t was, the slopes become more vertical and the cims less smooth.

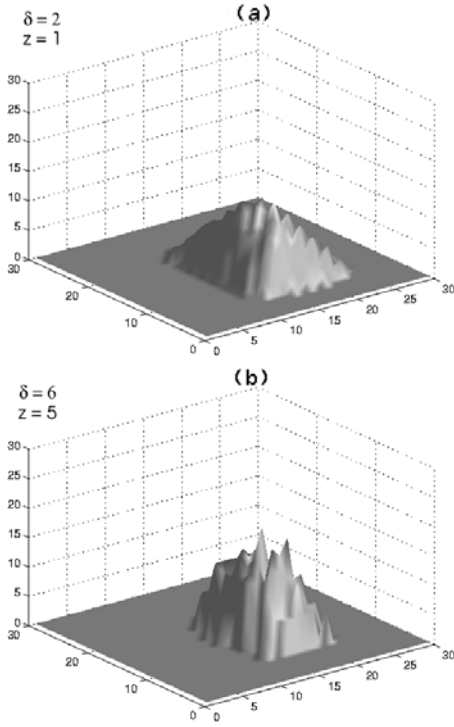


Figure 3: ESPM with $neigh_4$

One of the drawbacks of ASPM that have been resolved, according to our particular study, is that it is not possible to modify the visual aspect of the resulting pile by means of modifying the configuration parameters of the algorithm. Figures 1 and 2 will be always the same and it will not be possible to obtain different slopes nor smoother tops. In ESPM we can modify its parameters (δ_t and z_+) to obtain different kinds of piles. Figures 3 and 4 show the result of the simulation considering a neighbouring of 4 and 8 cells respectively. We can observe that the sand mountain is more realistic since the slopes are much more smooth.

The main difference among the two simulations using ESPM is that for the case of Figure 4 the surface is much more uniform because the sand is not going to the following cell in two-way traffic but in four.

Our tests exhibit a computational cost for every step of Algorithm 2 bigger than the observed in Algorithm 1. In both experiments, the cpu time elapsed for $neigh_4$ was 0.053s and for $neigh_8$ was 0.07s. However, the number of steps required to reach the stable state was 6 for $neigh_4$ and 5 for $neigh_8$, an order of magnitude lower than the used in the original algorithm, so it really outperforms its predecessor.

CONCLUSIONS AND FUTURE WORK

The main idea of this work was to verify how the ASPM model behaves when used to represent the dy-

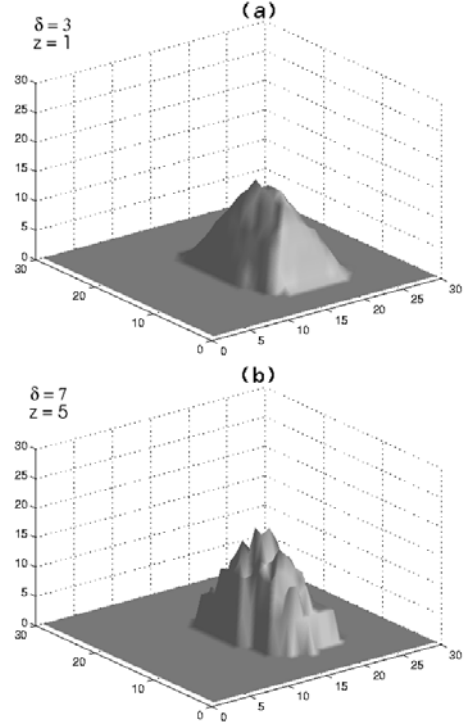


Figure 4: ESPM with $neigh_8$

namics of a sand pile that evolves until reaching its *final configuration*.

We observed that the ASPM model can not be used to visualize a sand pile simulation and a modification in the model has been proposed, called ESPM, which resembles more our objective.

The test done showed that our model has a computation time of the same order that the original ASPM and that is suitable to be used in applications requiring real time.

As future work some improvements will be done in ESPM for using at every step only those nodes that are affected by avalanches, not the whole mesh.

Another interesting line of research following this work raises when considering not only the fact that we are building a pile of sand but the opposite operation of removing material from it. In simulation applications, it is usual to interact with granular materials, adding, subtracting and also colliding with them, and all those effects modify the way the pile has to be viewed.

Another aspect that we have no mentioned is the possibility of simulating piles of mixed materials, with different cohesion properties. In this case, the ESPM algorithm will have to take into account not only the neighbourhood of the nodes but the classes of them.

REFERENCES

- Bak, P., Tand, C., and Wiesenfeld, K. (1988). Self-organized criticality. *Phys. Rev. A*, 38:364–374.
- Björner, A. and Lovász, L. (1992). Chip-firing games on directed graphs. *Journal of Algebraic Combinatorics*, 1:304–328.
- Brylawski, T. (1973). The lattice of integer partitions. *Discrete Mathematics*, 6:210–219.
- Dhar, D. (1990). Self-organized critical state of sand-pile automaton models. *Physical Review Letters*, 64(14):1613–1616.
- Dorso, C. O. and Dadamia, D. (2002). Avalanche prediction in abelian sandpile model. *Physica A: Statistical Mechanics and its Applications*, 308(1-4):179–191.
- Durand-Lose (1996). Grain sorting in the one-dimensional sand pile model. *COMPSYSTS: Complex Systems*, 10.
- Eric Goles, M. M. and Phan, H. D. (2002). The structure of a linear chip firing game and related models. *Theoretical Computer Science*, 270(1-2):827–841.
- Eriksson, K. (1993). *Strongly convergent games and coxeter groups*. PhD thesis, Kungl Tekniska Hogskolan.
- Goles, E. and Latapy, M. (2002). Complexity of grain-falling models. *Complexity International*.
- Latapy, M., Mantaci, R., Morvan, M., and Phan, H. D. (2001). Structure of some sand piles model. *Theoretical Computer Science*, 262(1-2):525–556.
- Moore, C. and Nilsson, M. (1999). The computational complexity of sandpiles. *Journal of Statistical Physics*, 96:205–224.
- Moss, S. (2001). Competition in internal markets: Statistical signatures and critical densities. Technical Report 01-79, Manchester Metropolitan University. UK.
- Novelli, J.-C. and Rossin, D. (2001). On the toppling of a sand pile. In *Discrete Models : Combinatorics, Computation, and Geometry*, pages 275–286. DM-CCG.
- Subramanian, R. and Scherson, I. (1994). An analysis of diffusive load-balancing. In *ACM Symposium on Parallel Algorithms and Architecture*, pages 220–225.

SIMULATION IN ELECTRONICS

CROSSTALK ANALYSIS ON PRINTED CIRCUIT BOARD BY MEANS OF WAVELETS

A. Kacha , F. Grenez , P. De Doncker

Service Electricité Générale, Faculté des Sciences Appliquées, Université Libre de Bruxelles, Av. F. D. Roosevelt 50,
CP 165/51, B-1050 Bruxelles, Belgium
Phone: +32-2-650-3087, Fax: +32-2-650-1347, E-mail: akacha@ulb.ac.be

K. Benmahammed

Département d'Electronique, Faculté des Sciences de l'Ingénieur, Université de Setif, Setif 19000, Algeria
Phone/Fax : +213-36-97-09, E-mail : khierben@ieee.org

KEYWORDS

Crosstalk analysis, electromagnetic interference, wavelets.

ABSTRACT

in the identification problems of the source lines of crosstalk on printed circuit boards, the requirements on the electromagnetic interference are specified in voltage which is converted to a maximum distance. Only tracks up to the specified maximum distance are considered for the purpose of disturbance line identification. Unfortunately, tracks beyond the specified maximum distance can result in large crosstalk. This paper proposes a wavelet-based approach which takes into account all tracks that are susceptible to produce interferences regardless the separating distance between the coupled tracks. A wavelet-decomposition of the disturbed signal allows to characterize the source of disturbance by means of the energy of the detail coefficients of the decomposition which makes easier the identification of the source of disturbance.

I. INTRODUCTION

The increase in speed of digital electronic circuits has led to several problems which may be taken into account in the earlier stage of design for the success of the system performance (Khan and Costache 1989). Fundamentally, three problems may be considered: Signal integrity, conducted and radiated emissions and susceptibility of the circuit (Gravelle and Wison 1992). Signal integrity is strongly affected by crosstalk which is defined as the unwanted interference signal generated by nearby tracks in the printed circuit board (PCB). As the switching speed increases and the PCBs miniaturized, the crosstalk between transmission lines imposes a significant signal integrity problem, mainly between the long parallel lines (Sohn et al. 2001). If the crosstalk is sufficiently large in magnitude and duration, it can result in a switching fault (Yeargan et al. 1988, Parker et al. 1994). Thus, it is of great importance to detect crosstalk using simulations during the design in order to eliminate or at least reduce it to acceptable levels before constructing the prototype.

One of the main problems that the designer may be faced to is determining when prototype failures are due to logic errors or to crosstalk. Reducing crosstalk effects allows to focus only on logic errors. Therefore, an efficient method for identifying the source line of crosstalk is required.

In geometric identification methods, crosstalk requirements are specified in voltage which is converted to a maximum distance. Only tracks up to the specified maximum distance are considered for the purpose of disturbance line identification. Unfortunately, tracks beyond the specified maximum distance can result in large crosstalk. Determining rules in geometric methods is an important issue because there are no general rules that allow to identify efficiently all tracks which may cause significant crosstalk. Recently, wavelets have proven to be useful in many problems related to crosstalk prediction and diagnostic of faults in transmission lines. In (Buccella and Orlandi 2000), wavelets has been applied to generate crosstalk signature in order to identify the faulty line and in (Antonini and Orlandi 2001), a wavelet packet decomposition has been used for the purpose of feature extraction and classification of the electromagnetic source of the disturbance. In this paper, we propose a wavelet-based approach for the purpose of disturbance source line identification. In the simulation, we consider two parallel tracks on a PCB separated by some distance. To the first track, we apply a high-frequency trapezoidal voltage characterized by small rise time and fall time and we predict the crosstalk generated in the second track. In fact, the propagating signal in the disturbed line is composed of the superposition of the induced crosstalk and the original signal without distortion. By performing a decomposition of the overall signal into coarse approximation and fine detail at different scales, it is possible to detect and quantify the crosstalk. As the distance between the two lines increases, the generated crosstalk decreases and this behaviour will be seen by inspecting the detail signal. Thus, by considering the detail in the wavelet decomposition, one can get an estimation of the distance between the source line of disturbance and the disturbed line. Examination of the crosstalk feature makes easier the identification of the different lines generating the electromagnetic

interferences. The paper is organized as follows. In section II, the crosstalk simulation is presented. In section III, we consider the problem of crosstalk analysis by means of a wavelet decomposition. In section IV, we give some examples to illustrate the wavelet approach in analysing crosstalk, and finally, conclusions are drawn in section V.

II. CROSSTALK SIMULATION

Several simulation programs for crosstalk computation have been developed (Weston 2001). To compute the crosstalk by means of SPICE simulation program, a numerical field solver program is required to obtain the electrical parameters that characterize the line. In this paper, the XFDTD simulation program has been used to compute crosstalk. The XFDTD simulator uses the method of finite difference time domain (FDTD) to solve the time-dependent Maxwell's equations (Pothecary and Railton 1991, Xiao et al. 2001). The curl equations are given by (Clayton 1999)

$$\vec{\nabla} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \quad (1)$$

$$\vec{\nabla} \times \vec{H} = \frac{\partial \vec{D}}{\partial t} + \vec{J} \quad (2)$$

where \vec{E} represents the electric field intensity vector, \vec{B} denotes the magnetic flux vector, \vec{H} represents the magnetic field intensity vector, \vec{D} represents the electric flux density vector and \vec{J} represents the current density vector. Equations (1) and (2) are solved at each time step. The three-dimensional simulation space is divided into elementary cubical elements known as Yee cells. The electric and magnetic fields are computed at points on a grid composed of the Yee cells.

In order to examine the crosstalk phenomenon, two parallel coupled tracks on a dielectric board characterized by a permittivity ϵ_r with a ground plane on the opposite side has been simulated (figure 1). In a typical situation, a source voltage $V_s(t)$ that has a resistor R_s is connected to a load R_L via the first track and the ground. Two terminations R_{NE} and R_{FE} are connected to the second track at the near end and the far end. The circuit associated to the source voltage is known as the generator circuit and the circuit corresponding to the resistors R_{NE} and R_{FE} is named the receptor circuit. The generator circuit will interact with the receptor circuit by generating electromagnetic fields giving rise to an induced crosstalk $V_{NE}(t)$ and $V_{FE}(t)$ in the near-end and far-end terminations of the receptor circuit. To illustrate the crosstalk phenomenon, consider two parallel tracks on a PCB with the following parameters: $\epsilon_r = 4.7$, $w = 2 \text{ mm}$, $s = 2 \text{ mm}$, $L = 10 \text{ cm}$ and $h = 1.5 \text{ mm}$. A trapezoidal source voltage (figure 2) with equal rise time and fall time of 0.25 ns , a frequency of 250 MHz , a duty cycle $\delta = 0.5$ and a source resistor of 50Ω was applied. The resistors R_L , R_{NE} and R_{FE} are equal to 50Ω . The predicted near-end and far-end crosstalk using XFDTD simulator are shown in figure 3 and figure 4, respectively. As shown, the near-end

crosstalk has a larger magnitude (0.4 V) and a longer duration than the far-end crosstalk. As the length of the parallel lines increases, the crosstalk becomes more important and can results in many problems related to the signal integrity.

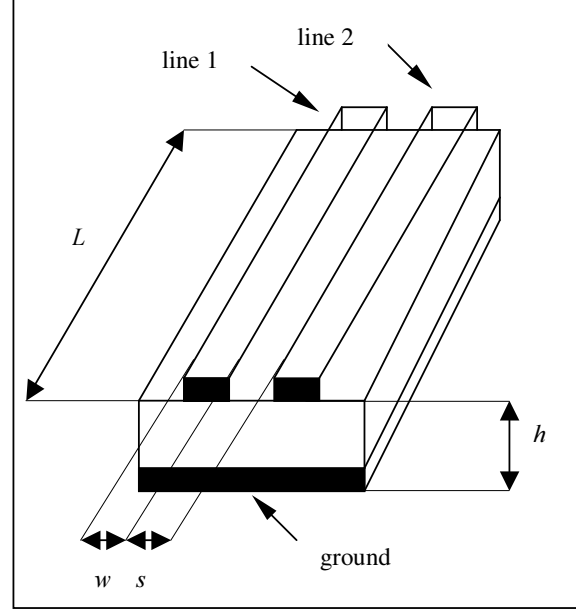


Figure 1: Two parallel coupled tracks on a PCB.

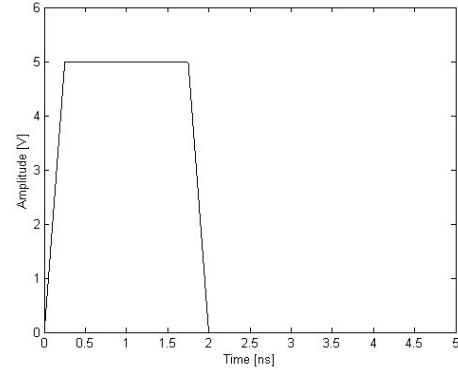


Figure 2: Disturbed Signal.

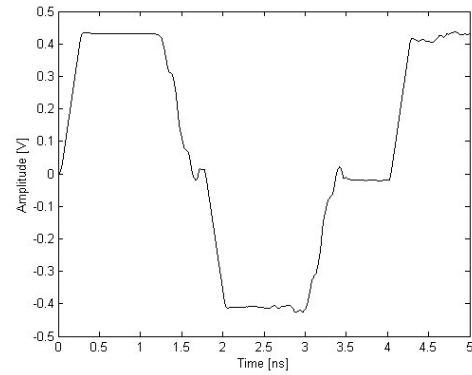


Figure 3: Near-end Crosstalk.

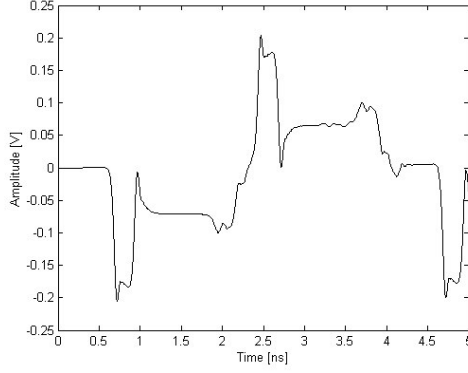


Figure 4: Far-end Crosstalk.

III. CROSSTALK ANALYSIS USING WAVELET DECOMPOSITION

The wavelet transform is a mathematical transformation that represents a signal in terms of shifted and dilated version of a single function called mother wavelet. A wavelet transform has a window whose bandwidth varies in proportion to the centre frequency of the wavelet. It provides the local scale of the signal over time. The continuous wavelet transform of a signal $f(t)$ with respect to some analysing wavelet ψ is defined as (Chui 1992)

$$Wf(b, a) = \int_{-\infty}^{\infty} f(t) \psi_{b,a}(t) dt \quad (3)$$

where

$$\psi_{b,a}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right), \quad a > 0$$

The parameters b and a are called translation and dilatation parameters, respectively. The normalization factor $a^{-1/2}$ is included to ensure $\|\psi_{b,a}\| = \|\psi\|$, where $\|\cdot\|$ denotes the norm. A necessary and sufficient condition for (3) to be invertible is that $\psi(t)$ satisfies the wavelet admissibility condition

$$\int_{-\infty}^{\infty} |\Psi(\omega)|^2 |\omega|^{-1} d\omega < \infty$$

where $\Psi(\omega)$ is the Fourier transform of $\psi(t)$. If $\psi(t)$ is sufficiently smooth and decay at infinity, which is usually the case, the admissibility condition can be written as

$$\int_{-\infty}^{\infty} \psi(t) dt = 0$$

The original signal can be reconstructed from its integral wavelet transform as

$$f(t) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} db \int_{-\infty}^{\infty} \frac{1}{a^2} [Wf(b, a)] \psi_{b,a}(t) da$$

where C_ψ is a constant that depends on the choice of wavelet and is given by

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty \quad (4)$$

The condition (4), known as the admissibility condition, restricts the class of functions that can be wavelets.

The continuous wavelet transform is a redundant representation. Frequently, we use its discrete version called the discrete wavelet transform (DWT) which is very useful in signal processing. In DWT, the scaling parameter a is taken to be of the form 2^j and b to be of the form $k 2^j$, where $k, j \in \mathbb{Z}$. With these values of a and b , the integral in (3) becomes

$$Wf(k 2^j, 2^j) = 2^{-j/2} \int_{-\infty}^{\infty} f(t) \psi(2^{-j} t - k) dt \quad (5)$$

Discretize now the function $f(t)$, and assume, for simplicity, the sampling rate to be 1, the integral in (5) can be written as

$$Wf(k 2^j, 2^j) = 2^{-j/2} \sum_n f(n) \psi(2^{-j} n - k)$$

One of the important properties of (10) is its time-variant nature. The DWT of a function shifted in time is quite different from the DWT of the original function.

There is a strong relationship between wavelet transform and multiresolution analysis (MRA). In the context of the MRA, one attempts to decompose a signal $f(t)$ as

$$f(t) = \sum_k u_k \phi_{j_0,k}(t) + \sum_{j=-\infty}^{j_0} \sum_k w_{j,k} \psi_{j,k}(t) \quad (6)$$

where j_0 is an integer, $\psi(t)$ and $\phi(t)$ are the wavelet and scaling functions, respectively, $w_{j,k}$ represent the wavelet or detail coefficients of $f(t)$ and u_k are called the scaling function or approximation coefficients. They are given by

$$w_{j,k} = \int_{-\infty}^{\infty} f(t) \psi_{j,k}(t) dt$$

$$u_k = \int_{-\infty}^{\infty} f(t) \phi_{j_0,k}(t) dt$$

In the representation given by (6), j indexes the scale or resolution of analysis and k indexes the spatial location of analysis. For a special choice of the wavelet $\psi(t)$ centred at time zero and frequency f_0 , the wavelet coefficient $w_{j,k}$ measures the signal content around time $2^j k$ and frequency $2^{-j} f_0$. The scaling coefficient u_k measures the local mean around time $2^j k$ and (6) may be interpreted as a multistage filtering decomposition of the signal $f(t)$ where the first term represents the low frequency

components and the second term represents the high frequency components of the signal (Mallat 1998).

In the context of crosstalk analysis, the signal in the disturbed line may be decomposed accordingly to (6) and the detail coefficients are analysed to get some information about the source line of disturbance. The mean energy of the detail coefficients of the wavelet decomposition provides a characterization of the source of disturbance and can be used to estimate the distance from the disturbed line to the source of disturbance.

IV. SIMULATION RESULTS

In this section, we give examples which apply the proposed approach based on a wavelet decomposition to crosstalk analysis. The configuration used in the simulation is that shown in figure 1 and the crosstalk were generated by means of XFDTD simulation program. In the first example, a trapezoidal source voltage with a rise time $t_r = 0.25 \text{ ns}$, a frequency $f = 250 \text{ MHz}$ and a duty cycle $\delta = 0.5$ was applied. A trapezoidal waveform with a rise time $t_r = 2.5 \text{ ns}$, a frequency $f = 50 \text{ MHz}$ and a duty cycle $\delta = 0.5$ is propagating down the disturbed line. In addition, it is assumed that the source voltage and the signal propagating down the disturbed line have the same phase. The wavelet decomposition of the undisturbed trapezoidal signal at level 2 using Daubechies wavelet of order 2 (db2) leads to a set of detail coefficients that are equal to zero except those corresponding to an abrupt change of the signal (Figure 5). For different tracks separations, the detail coefficients of the disturbed signal are shown in Figure 6. As seen, disturbance has introduced nonzero detail coefficients that have a magnitude comparable to the undisturbed signal. As the separation between the coupled lines increases, the magnitude of detail coefficients decreases. Thus, by analysing detail coefficients one can get an estimation of the distance separating the coupled line. As a characterization of the perturbation, the mean energy of detail coefficients may be used. As shown in Figure 7, the variation of the mean energy of the detail coefficients follows a roughly linear curve and then, one can use this energy to estimate the distance from the disturbed line to the source of disturbance.

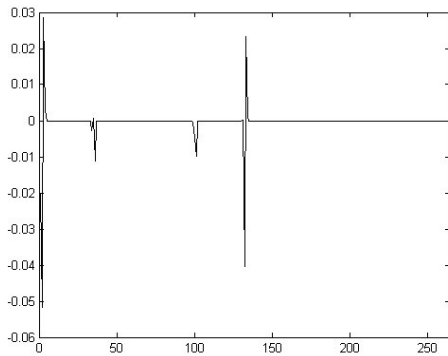
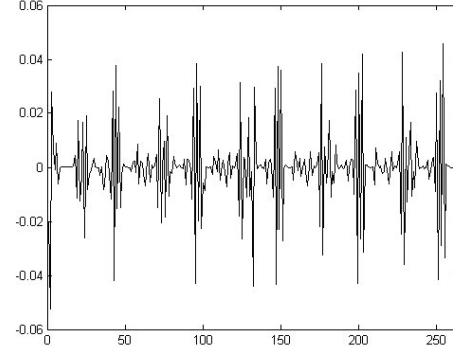
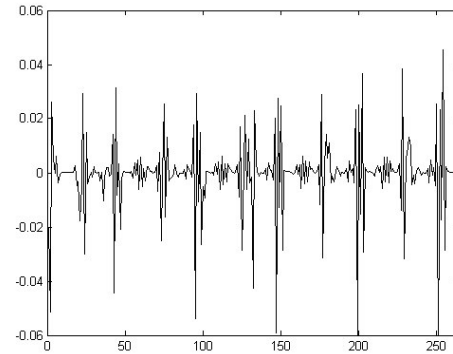


Figure 5: Detail coefficients of the undisturbed signal at level 2.

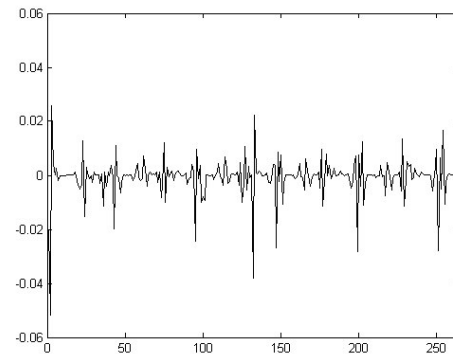
There is no rule for selecting the best level of resolution. In general, the choice is based on a priori knowledge of the signal. In our simulations, decomposition levels 2 and 3 lead to the best results. Notice that the analysis was done on a decimated version of the original signal generated by XFDTD simulator by a factor of 10 because it is unnecessary to take all samples.



(a)



(b)



(c)

Figure 6: Detail coefficients of the disturbed signal at level 2 using db2 wavelet for different values of the separating distance s . (a) $s = 1 \text{ mm}$. (b) $s = 5 \text{ mm}$. (c) $s = 10 \text{ mm}$.

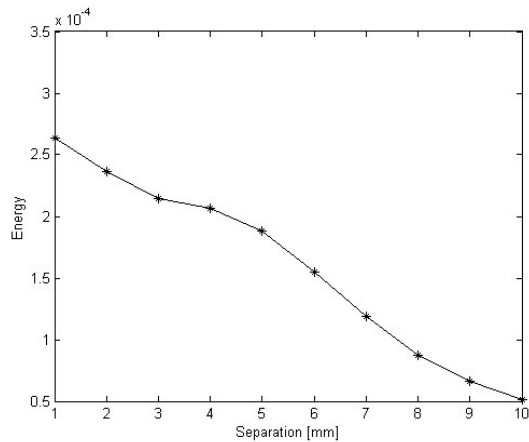


Figure 7: Energy of the detail coefficients versus distance separation between the two coupled lines.

V. CONCLUSION

A crosstalk analysis method based on wavelet decomposition has been proposed and tested on simulated signals propagating down two coupled tracks on a PCB. We have shown that a wavelet decomposition at level 2 of the disturbed signal allows to characterize the source of disturbance by means of the detail coefficients which makes easier the identification of the source of disturbance. This approach has the advantage to take into account all tracks that are susceptible to produce interferences without making any assumption about the separating distance between the coupled tracks. In the proposed approach, only one source of disturbance has been considered. A more complex situation that corresponds to several sources of disturbance may be addressed in the future research.

VI. REFERENCES

- Antonini, G. and A. Orlandi. 2001. "Wavelet Packet-Based EMI Signal Processing and Source Identification". *IEEE Transactions on Electromagnetic Compatibility*, Vol. 43, No. 2 (May), 140-148.
- Buccella, C. and A. Orlandi. 2000. "Diagnosing Transmission Line Termination Faults by Means of Wavelet Based Crosstalk Signature Recognition". *IEEE Transactions on Electromagnetic compatibility*, Vol. 23, No. 1 (Mar), 165-170.
- Chui, C. K. 1992. *An Introduction to Wavelets*. Academic Press, Boston.
- Clayton, R. P. 1992. *Introduction to Electromagnetic Compatibility*. John Wiley & Sons, New York.
- Gravelle, L. B. and P. F. Wison. 1992. "EMI/EMC in Printed Circuit Boards: A Literature Review". *IEEE Transactions on Electromagnetic compatibility*, Vol. 34, No. 2 (May), 109-116.
- Khan, R. L. and G. I. Costache. 1989. "Finite Element Method Applied to Modelling Crosstalk Problems on Printed Circuit Board". *IEEE Transactions on Electromagnetic compatibility*, Vol. 31, No. 1 (Feb), 5-16.
- Mallat, S. 1998. *A Wavelet Tour of Signal Processing*. Academic Press, San Diego.
- Parker, B. H.; A. K. Ray; and Z. Ghassemlooy. 1994. "Crosstalk in the Interconnection Bus for a High-Speed Digital Logic Circuit". *International Journal of Electronics*, Vol. 76, No. 2, 265-269.
- Pothecary, N. M. and C. J. Railton. 1991. "Analysis of Crosstalk on High-Speed Digital Circuits Using the Finite Difference Time-Domain Method". *International Journal of Numerical Modelling: Electronic Networks, Devices and fields*, Vol. 4, 225-240.
- Sohn, Y.; J. C. Lee; H. J. Park; and S. I. Cho. 2001. "Empirical Equations on Electrical Parameters of Coupled Microstrip Lines for Crosstalk Estimation in Printed Circuit Board". *IEEE Transactions on Advanced Packaging*, Vol. 24, No. 4 (Nov), 521-527.
- Weston, D. A. 2001. *Electromagnetic Compatibility: Principles and Applications*. Marcel Dekker.
- Xiao, F.; W. Liu; and Y. Kami. 2001. "Analysis of Crosstalk Between Finite-Length Microstrip Lines: FDTD Approach and Circuit Concept Modeling". *IEEE Transactions on Electromagnetic compatibility*, Vol. 43, No. 4 (Nov), 573-578.
- Yeargan, J. R.; R. L. Day.; and T. Nguyen. 1988. "Crosstalk in Multilayer Printed Circuit Boards". *IEEE Transactions on Education*, Vol. 31, No. 2 (May), 116-119.

ABDELLAH KACHA was born in Bouhatem, Algeria, in 1969. He received the degree of Engineer in 1993, from the university of Setif, Algeria and the degree of Magister in 1997, from the university of Batna, Algeria, both in electronic. Since 1998 he has been with the department of electronic engineering of the university of Jjel, Algeria. Since September 2003, he is working towards PhD in applied sciences at General Electricity Service of the Free University of Brussels, Belgium.

THREE-DIMENSIONAL MODELING OF THERMAL OXIDATION OF SILICON BY MEANS OF THE FINITE ELEMENT METHOD

Christian Hollauer

Hajdin Ceric

Siegfried Selberherr

Institute for Microelectronics, TU Vienna
Gusshausstr. 27–29, A-1040 Vienna, Austria
E-mail: Hollauer@iue.tuwien.ac.at

ABSTRACT. A numerical model which is suitable to describe three-dimensional thermal oxidation of silicon is proposed. By oxidation the three material components silicon, silicon dioxide and oxidant molecules are involved. The model takes into account that the diffusion of oxidants, the chemical reaction, and the volume increase occur simultaneously in a so-called reactive layer. This reactive layer has a spatial finite width, in contrast to the sharp interface between silicon and dioxide in the conventional formulation. The oxidation process is numerically described with a coupled system of equations for reaction, diffusion, and displacement. In order to solve the numerical formulation of the oxidation process the finite element scheme is applied.

INTRODUCTION

Thermal oxidation of silicon is one of the most important steps in fabrication of highly integrated electronic circuits, being mainly used for efficient isolation of adjacent devices from each other.

If a surface of a silicon body has contact with an oxidizing atmosphere, the chemical reaction of the oxidant (oxygen or steam) with silicon results in silicon dioxide. This reaction consumes silicon and the newly formed silicon dioxide has more than twice the volume of the original silicon. If a silicon dioxide domain is already existing, the oxidants diffuse through the oxide domain and react at the interface of oxide and silicon to form new oxide so that the dioxide domain is penetrated.

Thermal oxidation is a complex process where the three subprocesses oxidant diffusion, chemical reaction, and volume increase occur simultaneously. The volume increase is the main source of mechanical stress and strain, and these cause displacement (Zelenka 2000).

From the mathematical point of view the problem can be described by a coupled system of partial differential equations, one for the diffusion of the oxidant through the oxide, the second for the conversion of silicon into silicon dioxide at the interface, and a third for the mechanical problem of the Si–SiO₂–body, which can be modeled as an elastic, viscoelastic, or viscous body.

All published approaches can be classified essentially into three groups. The first type of method (Lorenz et al. 1985) maps the silicon dioxide domain in each time

step onto a simple numerical domain. The second approach uses the boundary element method for diffusion and displacement (Matsumota and Fukuma 1985). The third one (Rank and Weinert 1990) models the domain of computation by finite elements.

For a realistic and accurate oxidation simulation the three subproblems should be coupled, however, most oxidation models decouple them into a sequence of quasi-stationary steps. In our model all subprocesses are coupled simultaneously and the oxidation process can be simulated in three dimensions.

We will restrict the following explanation to the most simple physical model of linear oxidant diffusion and linear elastic displacement of the Si–SiO₂–body.

MODEL

We define a normalized silicon concentration

$$\eta(\vec{x}, t) = \frac{C_{Si}(\vec{x}, t)}{C_{0Si}} \quad (1)$$

where $C_{Si}(\vec{x}, t)$ is the silicon concentration at time t and point $\vec{x}(x, y, z)$ and C_{0Si} is the concentration in pure silicon. So η is 1 in pure silicon and 0 in pure silicon dioxide.

The oxidant diffusion is described by

$$D\Delta C(\vec{x}, t) = k(\eta)C(\vec{x}, t). \quad (2)$$

Here D is the diffusion coefficient and $k(\eta)$ is the strength of a spatial sink and not just a reaction coefficient at a sharp interface like in the standard model (Deal and Grove 1965). $k(\eta)C(\vec{x}, t)$ defines how many particles of oxygen per unit volume react in a unit time interval to silicon dioxide.

The change of η is described by

$$\frac{\partial \eta(\vec{x}, t)}{\partial t} = -\frac{1}{\lambda} k(\eta)C(\vec{x}, t)/N_1 \quad (3)$$

where λ is the volume expansion factor ($=2.25$) for the reaction from silicon to silicon dioxide, and N_1 is the number of oxidant molecules incorporated into one unit volume of silicon dioxide.

Furthermore, we define in (4) that $k(\eta)$ is linear proportional to η .

$$k = \eta(\vec{x}, t)k_{max} \quad (4)$$

The chemical reaction of silicon and oxygen causes a volume increase. The additional volume in a reference volume silicon ΔV , where we assume that the oxidant concentration C is constant, is given by

$$V^{add} = \frac{\lambda - 1}{\lambda} \Delta t \Delta V k(\eta) C(\vec{x}, t) / N_1. \quad (5)$$

We define the normalized additional volume V_{rel}^{add} as

$$V_{rel}^{add} = \frac{V^{add}}{\Delta V}. \quad (6)$$

For our model we assume, that the Si-SiO₂-body deforms elastically. In the theory of linear elasticity with small displacements $\vec{\theta}(x, y, z) = \{u(x, y, z) v(x, y, z) w(x, y, z)\}$ and strains ε_{ij} (i, j stands for x, y or z), the strain tensor $\tilde{\varepsilon}$ is defined as

$$\tilde{\varepsilon} = L_D \vec{\theta} \quad (7)$$

where $\vec{\theta}$ is the displacement vector and L_D is a differential operator, so that for example $\varepsilon_{xx} = \frac{\partial u}{\partial x}$ and $\varepsilon_{xy} = \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})$.

Assuming a linear material, the stress tensor $\tilde{\sigma}$ is given by

$$\tilde{\sigma} = \mathbf{D} \tilde{\varepsilon} \quad (8)$$

where \mathbf{D} is a (6x6) material matrix of elastic constants. The elastic constants are linear functions of Young's Modulus E and Poisson's ratio μ of the materials.

The force vector $\vec{f}(x, y, z) = \{f_x, f_y, f_z\}$ is the gradient of the stress tensor $\tilde{\sigma}$.

$$\vec{f} = \nabla \tilde{\sigma} \quad (9)$$

The most important part is that the volume expansion causes displacement. The normalized additional volume from (6) can be written as

$$V_{rel}^{add} = \varepsilon_x + \varepsilon_y + \varepsilon_z. \quad (10)$$

For an isotropic material the strain components are equal so that

$$\varepsilon_x = \varepsilon_y = \varepsilon_z = \frac{1}{3} V_{rel}^{add}. \quad (11)$$

With (7) and (11) the relationship between the volume expansion and the displacement is fully determined.

DISCRETIZATION

Weak Formulation

Before we start with the discretization we apply the weak formulation on (2) and (3).

So we apply the Galerkin method with linear test functions $N_k(\vec{x})$ on the diffusion equation from (2) and apply Green's theorem to remove the Laplace operator Δ as follows

$$\begin{aligned} k_{max} \int_V \eta C N_k dV &= D \int_V \Delta C N_k dV = \\ D \int_{\Gamma} \frac{\partial C}{\partial \vec{n}} N_k d\Gamma - D \int_V \nabla C \nabla N_k dV \end{aligned} \quad (12)$$

where $\frac{\partial C}{\partial \vec{n}} = 0$ and so the term $\int_{\Gamma} \frac{\partial C}{\partial \vec{n}} N_k d\Gamma$ is also zero, and so (12) is simplified to

$$k_{max} \int_V \eta C N_k dV = -D \int_V \nabla C \nabla N_k dV. \quad (13)$$

The application of the Galerkin method with the same linear test functions $N_k(\vec{x})$ to the distribution function from (3) leads to

$$\int_V \frac{\partial \eta}{\partial t} N_k dV = k_{max} \int_V \eta C N_k dV. \quad (14)$$

Oxidant Diffusion

In order to solve (2) and (3) on a three-dimensional domain with the volume V_{global} , we split the domain up into tetrahedral elements with the volume V and perform a finite element discretization. The spatial discretization for $C(\vec{x})$ on a single tetrahedral element is

$$C(\vec{x}, t = t_n) = \sum_{i=1}^4 c_i^{(t_n)} N_i(\vec{x}) \quad (15)$$

where $c_i^{(t_n)}$ is the oxidant concentration at node i and discrete time t_n . $N_i(\vec{x})$ is the form function on node i . The spatial discretization for $\eta(\vec{x})$ on a single tetrahedral element is

$$\eta(\vec{x}, t = t_n) = \sum_{i=1}^4 \eta_i^{(t_n)} N_i(\vec{x}) \quad (16)$$

where $\eta_i^{(t_n)}$ is the normalized silicon concentration at node i and discrete time t_n . $N_i(\vec{x})$ is the linear form function on a node i .

If we replace $C(\vec{x}, t)$ and $\eta(\vec{x}, t)$ in (13) with (15) and (16) we obtain

$$\begin{aligned} -D \int_V \left(\sum_{i=1}^4 c_i^{(t_n)} \nabla N_i \nabla N_k \right) dV = \\ k_{max} \int_V \left(\left(\sum_{i=1}^4 \eta_i^{(t_n)} N_i \sum_{i=1}^4 c_i^{(t_n)} N_i \right) N_k \right) dV = \\ k_{max} \int_V \left(\left(\sum_{i=1}^4 \eta_i^{(t_n)} c_i^{(t_n)} N_i \right) N_k \right) dV. \end{aligned} \quad (17)$$

With the following substitution

$$M_{ki} = \int_V N_k(\vec{x}) N_i(\vec{x}) dV \quad (18)$$

$$K_{ki} = \int_V \nabla N_k(\vec{x}) \nabla N_i(\vec{x}) dV \quad (19)$$

(17) is simplified to

$$\sum_{i=1}^4 (D K_{ki} c_i^{(t_n)} + k_{max} M_{ki} c_i^{(t_n)} \eta_i^{(t_n)}) = 0 \quad (20)$$

which is a non-linear equation system (k is the equation index) with the constants D , K_{ki} , k_{max} and M_{ki} and with the unknown variables $c_i^{(t_n)}$ and $\eta_i^{(t_n)}$ for one finite element.

Change of η

The spatial discretization for $C(\vec{x})$ and $\eta(\vec{x})$ is the same like in the last subsection and is already described by (15) and (16). Because of the time dependence of (3) an additional time discretization of the partial differential term $\frac{\partial \eta(\vec{x}, t)}{\partial t}$ is necessary. The time discretization is performed with the simple Backward-Euler method as

$$\frac{\partial \eta(\vec{x}, t = t_n)}{\partial t} = \frac{\eta(\vec{x}, t_n) - \eta(\vec{x}, t_{n-1})}{\Delta t} \quad (21)$$

where t_n and t_{n-1} are two successive discrete times.

If we replace $C(\vec{x}, t)$, $\eta(\vec{x}, t)$ and $\frac{\partial \eta(\vec{x}, t)}{\partial t}$ in (14) with the discrete expressions (15), (16) and (21), we obtain

$$\begin{aligned} \frac{1}{\Delta t} \int_V \left(\sum_{i=1}^4 (\eta_i^{(t_n)} - \eta_i^{(t_{n-1})}) N_i N_k \right) dV = \\ k_{max} \int_V \left(\sum_{i=1}^4 (\eta_i^{(t_n)} c_i^{(t_n)}) N_i N_k \right) dV. \end{aligned} \quad (22)$$

With the substitution (18) the last equation is simplified to a non-linear equations system (k is the equation index)

$$\sum_{i=1}^4 \left(M_{ki} (\eta_i^{(t_n)} - \eta_i^{(t_{n-1})}) - k_{max} c_i^{(t_n)} \eta_i^{(t_n)} \right) \frac{1}{\Delta t} = 0 \quad (23)$$

with the unknown variables $c_i^{(t_n)}$ and $\eta_i^{(t_n)}$ and with the constants M_{ki} , k_{max} and $\frac{1}{\Delta t}$ for one finite element. The values for $\eta_i^{(t_{n-1})}$ are already determined at the previous time step.

If we combine the two equation systems (20) and (23), we obtain a non-linear equations system for one finite element (with eight equations and the eight unknowns $c_{1..4}^{(t_n)}$ and $\eta_{1..4}^{(t_n)}$). Now we are able to solve the system (for example with the Newton method) at each time point t_n and the values for $c_i^{(t_n)}$ and $\eta_i^{(t_n)}$ can be determined.

Mechanics

The finite element discretization for a mechanical system has been already often described, for example by (Zienkiewicz 1987). Because of this fact we will restrict this subsection only to some steps which are important for the oxidation simulation.

After discretization of the continuum, the relationship between strain and displacement (7) can be written as

$$\tilde{\varepsilon}^e = \mathbf{B} \vec{d}^e = [\mathbf{B}_i, \mathbf{B}_j, \mathbf{B}_m, \mathbf{B}_p] \quad (24)$$

in which $\tilde{\varepsilon}^e$ is the strain tensor, \vec{d}^e is the displacement

vector and i, j, m and p are the four nodes on a single tetrahedron.

The element displacement is defined by the 12 displacement components of the nodes as

$$\vec{d}^e = \begin{Bmatrix} \vec{d}_i \\ \vec{d}_j \\ \vec{d}_m \\ \vec{d}_p \end{Bmatrix} \quad \text{with} \quad \vec{d}_i = \begin{Bmatrix} u_i \\ v_i \\ w_i \end{Bmatrix} \quad \text{etc.} \quad (25)$$

The submatrix \mathbf{B}_i for the node i is

$$\mathbf{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x}, & 0, & 0 \\ 0, & \frac{\partial N_i}{\partial y}, & 0 \\ 0, & 0, & \frac{\partial N_i}{\partial z} \\ \frac{\partial N_i}{\partial y}, & \frac{\partial N_i}{\partial x}, & 0 \\ 0, & \frac{\partial N_i}{\partial z}, & \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial z}, & 0, & \frac{\partial N_i}{\partial x} \end{bmatrix} = \begin{bmatrix} b_i, & 0, & 0 \\ 0, & c_i, & 0 \\ 0, & 0, & d_i \\ c_i, & b_i, & 0 \\ 0, & d_i, & c_i \\ d_i, & 0, & b_i \end{bmatrix} \quad (26)$$

with the linear form function $N_i(\vec{x})$ defined as

$$N_i(\vec{x}) = a_i + b_i x + c_i y + d_i z \quad (27)$$

in which a_i , b_i , c_i and d_i are constant geometrical coefficients for the finite element. For example b_i is

$$b_i = -\det \begin{vmatrix} 1, & y_j, & z_j \\ 1, & y_m, & z_m \\ 1, & y_p, & z_p \end{vmatrix}. \quad (28)$$

The entire inner virtual work on a finite element is

$$W_{inner} = \int_V \{\tilde{\varepsilon}^e\}^T \sigma^e dV \quad (29)$$

in which the transposed strain tensor is

$$\{\tilde{\varepsilon}^e\}^T = \vec{d}^{eT} \mathbf{B}^T \quad (30)$$

and the stress tensor (8) can be written as

$$\sigma^e = \mathbf{D} \tilde{\varepsilon}^e = \mathbf{D} \mathbf{B} \vec{d}^e. \quad (31)$$

That leads us to the following equation for W_{inner} .

$$W_{inner} = \vec{d}^{eT} \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} \vec{d}^e dV \quad (32)$$

The outer virtual work on a finite element, caused by the node forces is

$$W_{outer} = \vec{d}^{eT} \vec{f}^e. \quad (33)$$

On a element the inner work must be equal with the outer work.

$$W_{inner} = \vec{d}^{eT} \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} \vec{d}^e dV = \vec{d}^{eT} \vec{f}^e = W_{outer} \quad (34)$$

With the substitution

$$\mathbf{K}^e = \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV \quad (35)$$

where \mathbf{K}^e is the so-called stiffness matrix, we obtain a linear equation system for the mechanical problem.

$$\mathbf{K}^e \vec{d}^e = \vec{f}^e \quad (36)$$

The most important part is, how the volume increase (5), caused by the chemical reaction of silicon to silicon dioxide, loads the displacement problem.

Due to (11) we obtain the components ε_x^e , ε_y^e and ε_z^e for the strain tensor $\tilde{\varepsilon}^e$ and with

$$\vec{f}_i^e = -\mathbf{B}_i^T \mathbf{D} \tilde{\varepsilon}^e V^e \quad (37)$$

the relationship between the volume expansion and the node forces is given, and with (36) and (37) the displacements on the nodes is fully determined.

By coupling (20), (23) and (36), a local equation system for one finite element is given, which is a complete numerical formulation of the oxidation process with its oxidant diffusion, chemical reaction and volume increase at any time.

SIMULATION PROCEDURE

In the first step of the simulation procedure, we perform a finite element discretization. With this aim in view we split up the Si-SiO₂-body into tetrahedral elements and that results in a tetrahedral grid on the domain. The size of the tetrahedrons and, as a result of that, the number of the finite elements can be influenced by the meshing module.

In the next step we set the initial values for the oxidant concentration C and the normalized silicon concentration η on the grid nodes. For example η must be 1 in a pure silicon domain.

Since the oxidation process is time dependent, the actual simulation time must be reset at the beginning of the simulation.

As shown in Fig. 1, we iterate over all finite elements and build the local equation system for one element for every actual discrete time. The local system describes the oxidation process numerically for one element with the coupled system for diffusion, chemical reaction and the displacement problem. Note, that it would be wrong to solve the relative simple local equation system for one element. The finite element method includes the superposition principle but not in the way to add up locally calculated results in order to determine the global results. In our case "global" has a spatial meaning and stands for the whole discretized domain.

In order to describe the global oxidation process we need a global coupled equation system. The components of the global equation system are assembled from the local equation system by using the superposition principle.

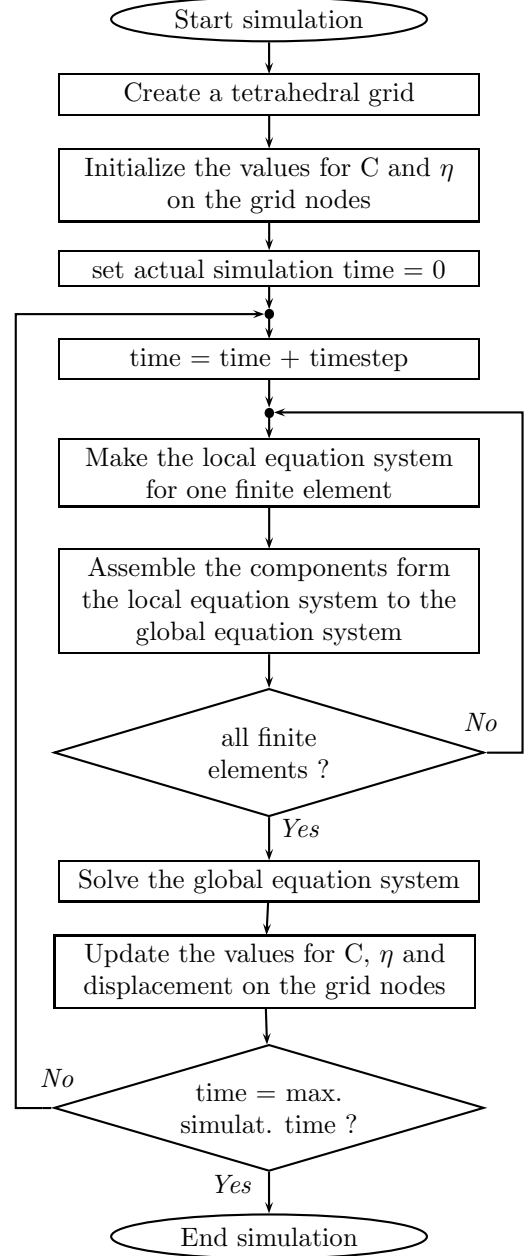


Figure 1: Simulation procedure

After the iteration over all elements is finished, the global assembled equation system is also finished. Now the global non-linear equation system can be solved and we obtain the results for the C , η and displacement values for the global discretized oxidation process for the actual time step.

With these results we update the values for C , η and displacement on the grid nodes by adding up the new results to the already existing values on the grid nodes and so the values for C , η and displacement always keep pace with the actual simulation time.

If the above described procedure is finished, we increase the actual simulation time and start with the assembling loop again as long as the actual time is equal to the maximum simulation time.

A DEMONSTRATIVE EXAMPLE

As example a silicon body with the initial dimension $(0.5 \times 0.4 \times 0.5) \mu\text{m}$ as shown in Fig. 2 is oxidized. For the simulation the following parameters were chosen:

$$C^* = 3 \cdot 10^7 \left[\frac{\text{part.}}{\mu\text{m}^3} \right], \quad D = 0.08 \left[\frac{\mu\text{m}^2}{\text{s}} \right], \quad k_{\max} = 40 \left[\frac{1}{\text{s}} \right].$$

Here C^* is the surface oxidant concentration.

In this example only the upper surface of the body has contact with the oxidizing atmosphere. As shown in Figs. 2–4 the bottom surface is fixed and on the rest of the surfaces there are free mechanical boundary conditions applied.

In the Figs. 2–4 the angel of view is always the same and the proportions of the body geometry are also true, so that the displacement effects caused by the volume increase can be watched correct.

The colour in the figures shows the value of the normalized silicon concentration. This means that blue is pure silicon dioxide and red is pure silicon and the other colours are the reaction layer.

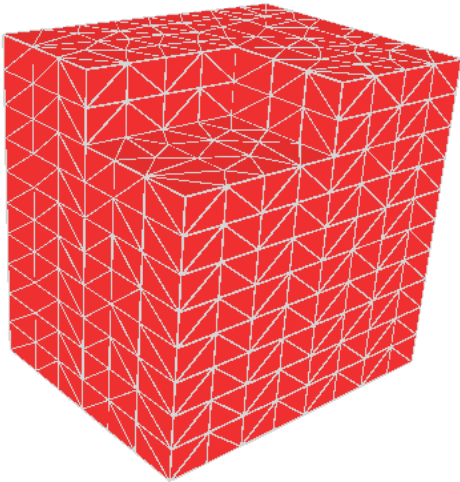


Figure 2: Initial silicon body before oxidation.

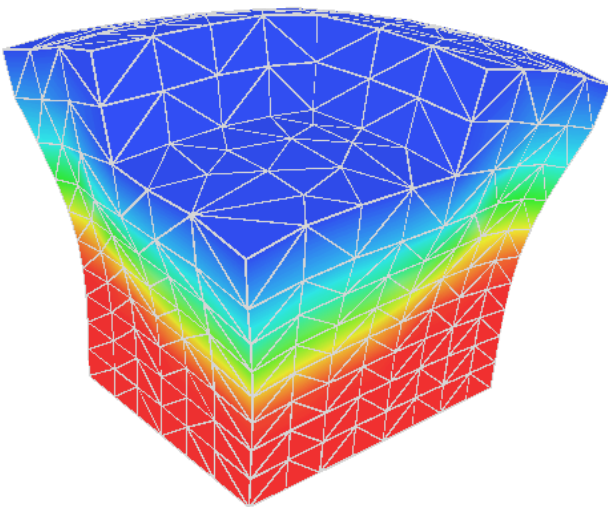


Figure 3: Deformation and silicon dioxide distribution (blue region) at some time t_1 .

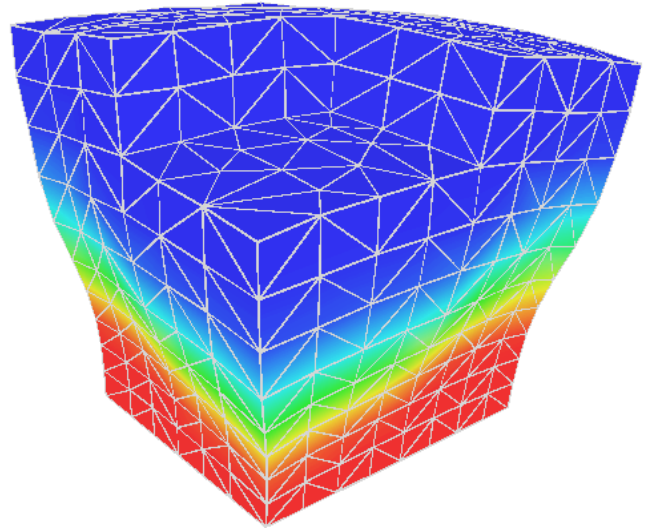


Figure 4: Deformation and silicon dioxide distribution (blue region) at time $2 * t_1$.

CONCLUSIONS

A three-dimensional oxidation model which is based on the finite element technique has been proposed. In this model it is assumed that the interface between the silicon and oxide is a reaction layer with finite width instead of a sharp interface. In this layer there is a mixture of the three components silicon, oxidants, and oxide.

One of the advantages of this model is that the numerical formulation, consisting of a coupled differential equation system, describes the complete physical oxidation process in a very realistic way.

As demonstrated on a numerical example, this model is a powerful tool to simulate the whole oxidation process on three-dimensional semiconductor structures.

REFERENCES

- Deal B.E. and Grove A.S. 1965. "General Relationship for the Thermal Oxidation of Silicon". *Journal Applied Physics*, vol. 36, no. 12, 3770–3778.
- Lorenz J., Pelka J., Sachs A., Seidl A., and Svoboda M. 1985. "Composite – A complete Modeling Program for Silicon Technology". *IEEE Transactions on CAD*, vol. 4, no. 4, 421–430.
- Matsumoto H. and Fukuma M. 1985. "Numerical Modeling of Nonuniform Si Thermal Oxidation", *IEEE Transactions on Electron Devices*, vol. 32, no. 2, 132–140.
- Rank E. and Weinert U. 1990. "A Simulation System for Diffuse Oxidation of Silicon: A Two-Dimensional Finite Element Approach". *IEEE Transactions on CAD*, vol. 9, no. 5, 543–550.
- Zelenka S. 2000. "Stress Related Problems in Process Simulation". PhD thesis. Federal Institute of Technology, Zurich, Switzerland.
- Zienkiewicz O.C. 1987. *The Finite Element Method, volume 1: Basic Formulation and Linear Problems*, 4th edition, McGraw - Hill, Maidenhead, England.

STATISTICAL ANALYSIS FOR THE THREE-DIMENSIONAL MONTE CARLO SIMULATION OF ION IMPLANTATION

Robert Wittmann

Andreas Hössinger

Siegfried Selberherr

Institute for Microelectronics, TU Vienna
Gusshausstr. 27–29, A-1040 Vienna, Austria
E-mail: Wittmann@iue.tuwien.ac.at

KEYWORDS

Ion Implantation, Monte Carlo Method, Statistical Fluctuation of the Doping Profile

ABSTRACT

Without a proper statistical analysis of the simulation output data, it is not possible to assess the statistical accuracy of three-dimensional Monte Carlo simulation results. The Monte Carlo technique applied to the simulation of ion implantation produces a statistical fluctuation of the doping profile, in particular in the three-dimensional case. The statistical accuracy is determined basically by the number N of simulated ion trajectories. It depends also on the variation of the ion concentration up to several orders of magnitudes in the simulation domain. The theoretical simulation error of order $1/\sqrt{N}$ has been expectedly verified by several simulation experiments with different N . The paper describes the application of statistical methods in order to evaluate the accuracy of three-dimensional ion implantation results compared to one-dimensional results. We propose a method to determine the number of trajectories required to obtain a specified precision in a three-dimensional Monte Carlo simulation study.

INTRODUCTION

Ion implantation is the state-of-the-art method for doping semiconductors because of its high controllability. The small dimensions of modern semiconductor devices have led to simulation applications which require a high accurate and full three-dimensional treatment. Since the process of ion implantation has a statistical nature, it is straightforward to use statistical methods to simulate it on computers. The most important of such methods is the Monte Carlo method which is based on applying random behavior at an atomistic level (Hobler and Selberherr 1989), (Ziegler et al. 1995).

Particularly, the position where an ion hits the crystalline target is calculated using random numbers. Furthermore, the lattice atoms of the target are in permanent movement due to thermal vibrations. Thus, the

actual positions of the vibrating atoms in the target are also simulated using random numbers. The trajectory of each implanted ion is determined by the interactions with the atoms and the electrons of the target material. The final position of an implanted ion is reached where it has lost its complete energy. The accuracy of the simulation is mainly determined by the complexity of the models that describe the physical behavior. These models are applicable for a wide range of implantation conditions without additional calibration. The number of simulated ions must be considerably increased in order to achieve the same statistical accuracy for three-dimensional simulations as in two dimensions. Therefore the computational effort grows approximately proportional to the surface area of the simulation domain. A very common mode of operation is to simulate an arbitrary large number N of ion trajectories and then treat the resulting ion concentration estimates as the exact doping profile. In spite of the use of an expensive simulation model misleading results might be obtained, if the random nature of the output data is ignored. From our point of view no in-depth analysis of the simulation accuracy of Monte Carlo process simulations has been carried out so far, and in this work we will present the first comprehensive investigation of the statistical accuracy for three-dimensional Monte Carlo simulations of ion implantation.

The practitioner of a Monte Carlo simulation is always concerned with the computational time and the statistical accuracy of the simulation. Both are related to the simulation rate of convergence to the "true" value. The standard error in the simulation can be viewed as the standard deviation of the random sample divided by an increasing function of N , the number of simulated ions. We assume that all simulated ions are statistically independent. One way to reduce the simulation error is by using a smart postprocessing of the raw data. The statistical fluctuation can be reduced effectively by smoothing the Monte Carlo simulation results in a postprocessing step (Heitzinger et al. 2003). The other obvious way to reduce the error is by increasing the number N of simulated ions. The traditional Monte Carlo technique using pseudo random numbers has only a convergence rate of order $1/\sqrt{N}$, which follows from the Central Limit

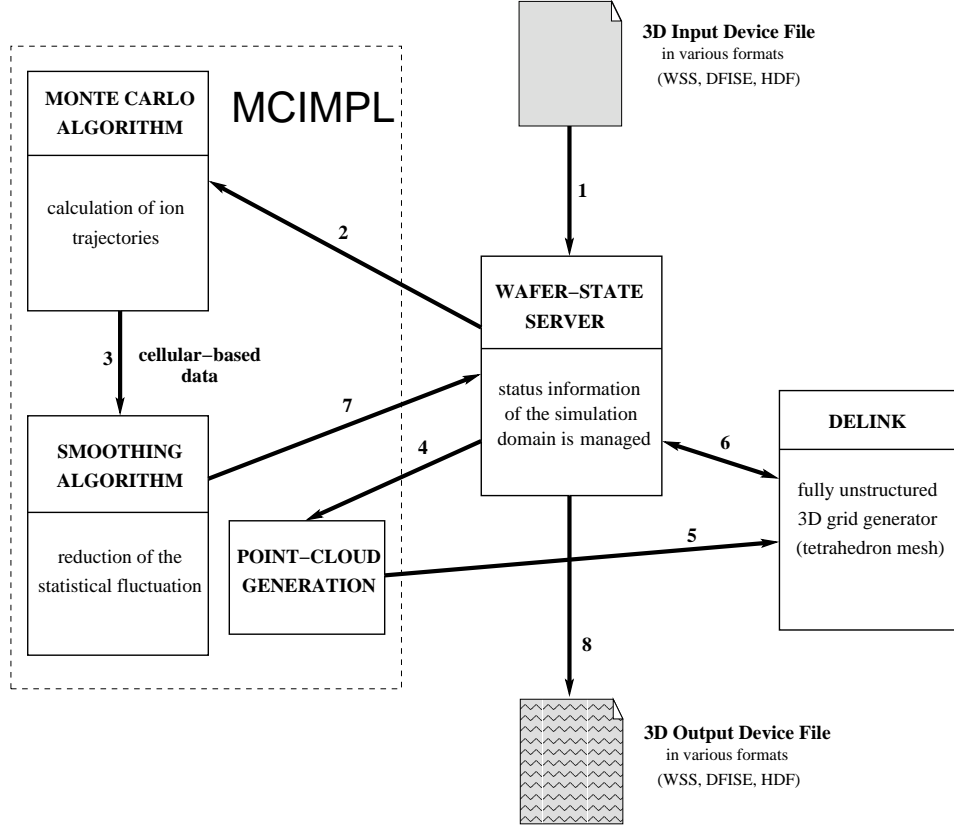


Figure 1: Data flow and involved process simulation tools.

Theorem (Law and Kelton 2000). This rate is independent of the dimension and depends only on the number N of simulations.

However, there is always a trade-off between the computational effort and the simulation error. In particular with regard to three-dimensional Monte Carlo simulations additional speed-up techniques have to be used in order to get reasonably low statistical noise by practicable long simulation runs. Examples of such speed-up techniques are the trajectory split method and the trajectory reuse method.

THE SIMULATOR

All Monte Carlo simulation experiments were performed with the object-oriented, multi-dimensional ion implantation simulator MCIMPL. The simulator is based on a binary collision algorithm and can handle arbitrary three-dimensional device structures consisting of several amorphous materials and crystalline silicon. In order to optimize the performance, the simulator uses cells arranged on an ortho-grid to count the number of implanted ions and of generated point defects. The final concentration values are smoothed and translated from the internal ortho-grid to an unstructured grid suitable for subsequent process simulation steps, like finite element simulations for annealing processes.

Figure 1 shows the data flow during the simulation of

ion implantation. The simulator MCIMPL is embedded in a process simulation environment by using the object-oriented WAFER-STATE SERVER library (Binder and Selberherr 2003).

The WAFER-STATE SERVER has been developed in order to integrate several three-dimensional process simulation tools used for topography, ion implantation, and annealing simulations. It holds the complete information describing the simulation domain in a volume mesh discretized format, and it provides convenient methods to access these data. The idea was that simulators make use of these access methods to initialize their internal data structure, and that the simulators report their modifications of the wafer structure to the WAFER-STATE SERVER. Thereby a consistent status of the wafer structure can be sustained during the whole process flow.

The meshing strategy of DELINK follows the concept of advancing front Delaunay methods and produces tetrahedral grid elements (Fleischmann and Selberherr 2002).

ANALYSIS METHOD

For the analysis of three-dimensional simulation output, several numerical experiments were performed on a three-dimensional structure equivalent to a one-dimensional problem. In particular, implantations of phosphorus ions into a crystalline silicon substrate were

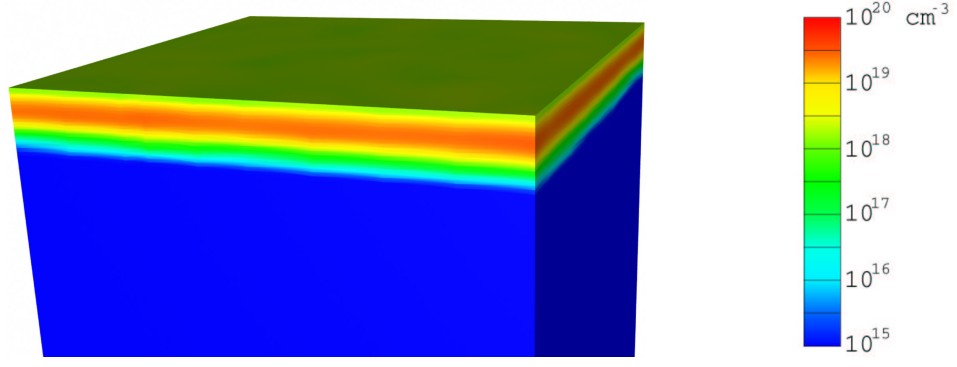


Figure 2: Accurate Monte Carlo simulation result of phosphorus implantation in silicon with $N = 10^7$ simulated ions, an energy of 25 keV, and a dose of 10^{14} cm^{-2} .

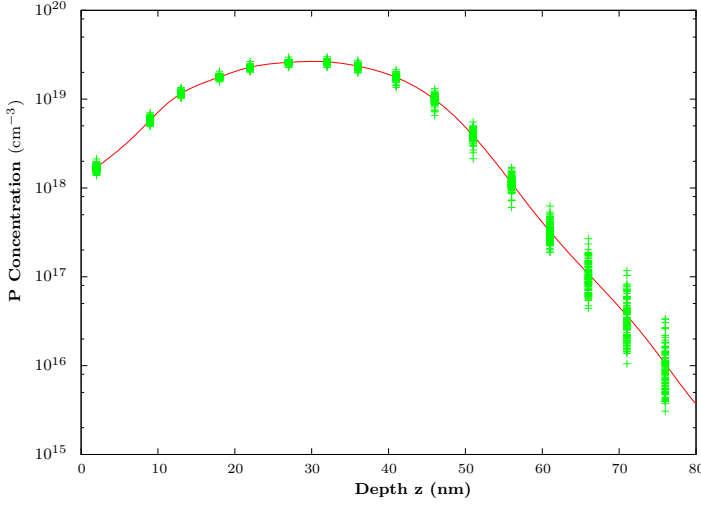


Figure 3: Variability of the three-dimensional result.

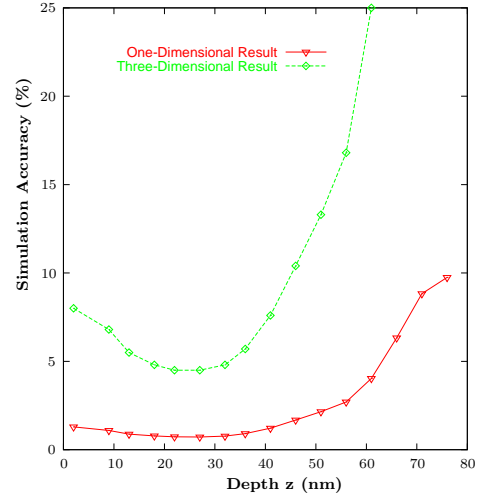


Figure 4: Result evaluation.

simulated with different N . Figure 2 shows the three-dimensional result for an accurate simulation with $N = 10^7$ ions. We extracted the z coordinates and the phosphorus concentration values C (vertical direction) from all 2972 grid points of the unstructured grid. This leads to Figure 3 which demonstrates the statistical fluctuation of the impurity concentration at equal penetration depth z .

The relative standard deviation of the impurity concentration in a plane $z = \text{const}$ is a measure for the simulation error of three-dimensional results compared to one-dimensional results. The mean impurity concentration $\bar{C}(n)$ of n grid points at equal location z forms the one-dimensional doping profile. The standard deviation $S(n)$ of a sample defined by the concentration values of n grid points in a plane $z = \text{const}$ is given by

$$S(n) = \sqrt{\frac{\sum_{i=1}^n [C_i - \bar{C}(n)]^2}{n-1}} \quad (1)$$

$$\sigma = \frac{S(n)}{\bar{C}(n)} \quad (2)$$

The relative standard deviation σ according to (2) is calculated in order to evaluate the three-dimensional result.

Figure 4 demonstrates the statistical accuracy of the three-dimensional result related to the one-dimensional doping profile. Most of the simulated ions come to rest close to the mean projected range R_p , causing a small variance there. Due to the very low dopant concentration in deeper regions (typically more than 10^4 times lower than at the maximum), insufficient events lead to an increase of the statistical noise.

Being based on random numbers, the results obtained with the Monte Carlo technique are never exact, but rigorous in a statistical sense. The results converge to the used model characteristics. A 90% confidence interval is constructed for the mean, in order to assess the relative error of the one-dimensional doping profile in relation to the model limit value. The half of the approximate 90% confidence interval, $\Delta(n)$, using the t distribution (Law and Kelton 2000) is given to

$$\Delta(n) = t_{n-1, 0.95} \frac{S(n)}{\sqrt{n}} \quad (3)$$

The relative statistical error $\epsilon(n)$ for the one-dimensional doping profile can be defined as

$$\epsilon(n) = \frac{\Delta(n)}{\bar{C}(n)} \quad (4)$$

The assessed statistical accuracy of the one-dimensional doping profile according to (4) is also demonstrated in Figure 4.

The accuracy of the Monte Carlo result is determined by the number of counted ions per cell. The distribution of N ions determines the one-dimensional doping profile by using a scaling factor α :

$$\alpha \int_0^\infty \bar{C}(z) dz = N \quad (5)$$

(5) can be used in order to calculate the factor α by means of numerical integration. For a small volume of the width Δz (cell dimension) the local number N_i of simulated ions is determined by

$$N = \sum_i N_i, \quad N_i = \alpha \bar{C}_i \Delta z \quad (6)$$

The division by all cells of a z plane yields to the average ions per cell, which is demonstrated in Figure 5 for $N = 10^7$ simulated ions. Each bar is located at the grid points of the internal ortho-grid. The histogram demonstrates that in deep regions only one simulated ion per cell is available in the mean. More and more empty cells at increasing penetration depth downgrade the statistics dramatically.

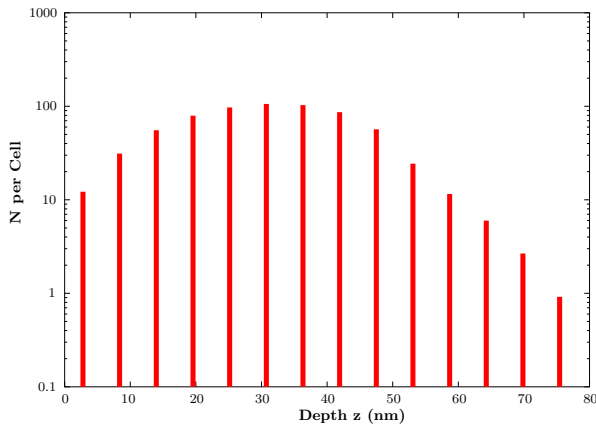


Figure 5: Estimated simulated ions per cell for a total number $N = 10^7$.

An essential contribution to the accomplished accuracy of the final result is obtained through the reduction of the statistical fluctuation by an implemented smoothing algorithm (Heitzinger et al. 2003). This algorithm sweeps a small rectangular grid over the points of the new tetrahedral grid and uses an approximation by generalized Bernstein polynomials. The Bernstein approximation of a concentration value on a new grid point by using the values of cells located close to the new grid point reduces significantly the statistical noise. The bad statistics generated by empty cells can be attenuated by averaging the values of surrounding cells.

We extracted again z coordinates and phosphorus concentration values from all $120 \times 112 \times 20$ cells of the simulation area. Figure 6 compares the relative standard

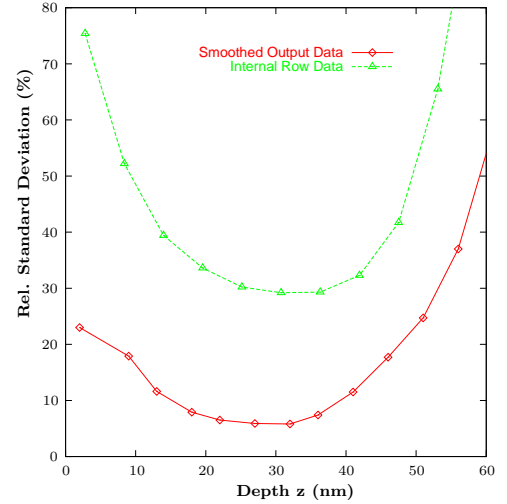


Figure 6: Improvement of the statistical accuracy by smoothing ($N = 10^6$).

deviation for $N = 10^6$ ions before and after smoothing. Thus a significant improvement of the statistical accuracy of Monte Carlo results can be achieved through the filter effect of the Bernstein polynomials, which eliminates high-frequency fluctuations from the original data.

Of great importance for the simulation is the weight of an ion, which is defined by the ratio of the number of real ions N_{real} to the number of simulated ions N .

$$\text{Weight} = \frac{N_{\text{real}}}{N} \quad (7)$$

In our simulation experiment shown in Figure 2 the surface dimension is $0.7\mu\text{m} \times 0.65\mu\text{m}$. With a dose of 10^{14} cm^{-2} , 455000 ions are implanted. With 10^7 simulated ions the weight of an ion results to 0.0455. In practice the real-world implanted doping profile has also a fluctuation due to the statistical nature of the implantation process. In our simulation example a real ion has only a very little weight. Thus the simulation result can be considered as a simulation of averaging over multiple real-world implantations.

IMPROVEMENT OF THE SIMULATOR

The crucial factor for the duration and accuracy of the simulation is the specified number N of ion trajectories as input data of the simulator. One drawback of the fixed-sample-size procedure based on N simulated ions is that the analyst has no immediate control over the precision of the output data. We suggest an improvement of the used fixed-sample-size procedure by determining the duration of the simulation also through a specified precision as input data of the simulator.

The simulation error of the Monte Carlo method is of order $1/\sqrt{N}$. The relationship between the standard deviation σ and the number N of ions is given by

$$\sigma = \text{const} \cdot \frac{1}{\sqrt{N}} \quad (8)$$

This relationship has been expectedly verified by simulation experiments with different N and is demonstrated in Figure 7. It can also be used to assess the number of trajectories required to obtain a specific precision in a Monte Carlo simulation study.

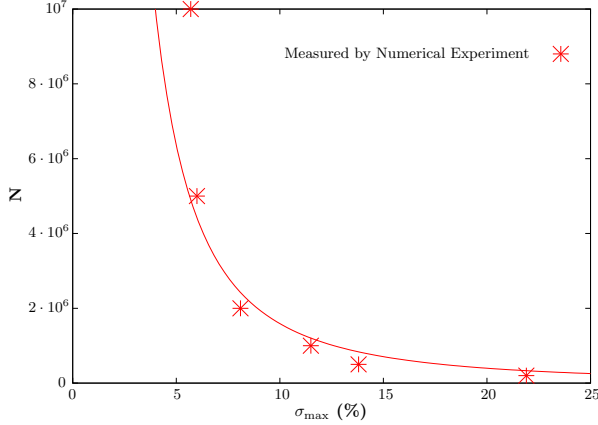


Figure 7: Required N as a function of the desired three-dimensional accuracy .

As measure of the simulation accuracy, the desired maximum of the relative standard deviation σ_{\max} within the range $2 \cdot \Delta R_p$ (twice the straggling at the mean projected range) of the doping profile is used. In our experiment of Figure 2, $2 \cdot \Delta R_p = 22$ nm at $R_p = 30$ nm. For the calculation of the required N as function of the given standard deviation σ_{\max} , a parameter γ is used which takes the incident atom species and the implantation energy into account. The following formula can be used to assess the N for a specified surface area A and a desired precision σ_{\max} :

$$N = \gamma \frac{A}{A_0} \frac{1}{\sigma_{\max}^2}, \quad A_0 = 0.455 \mu\text{m}^2 \quad (9)$$

Figure 7 demonstrates this relationship for a phosphorus implantation, an ion energy of 25keV, $A = A_0$, and parameter $\gamma = 15992$.

CONCLUSION

The functionality of the three-dimensional Monte Carlo simulator MCIMPL for ion implantation is demonstrated. The statistical fluctuation of the simulation result caused by the stochastic simulation method and the expensive three-dimensional treatment are analyzed. The evaluation of the statistical accuracy for three-dimensional results is performed by the use of statistical methods like calculating the standard deviation or the confidence interval of the output data. The gained insight into the relationships responsible for the statistical accuracy is used in order to achieve a better controllability of the simulator.

REFERENCES

- Binder T. and Selberherr S. 2003. "Rigorous Integration of Semiconductor Process and Device Simulators". *IEEE Transactions on CAD*.
- Fleischmann P. and Selberherr S. 2002. "Enhanced Advancing Front Delaunay Meshing in TCAD". *SISPAD 2002*
- Heitzinger C., Hössinger A. and Selberherr S. 2003. "An Algorithm for Extracting and Smoothing Three-Dimensional Monte Carlo Simulation Results". *IEEE Transactions on CAD*.
- Hobler G. and Selberherr S. 1989. "Monte Carlo Simulation of Ion Implantation into Two- and Three-Dimensional Structures". *IEEE Transactions on CAD*, Vol. 8, No. 5, 450-489.
- Law A.M. and Kelton W.D. 2000. "Simulation Modeling and Analysis". McGraw-Hill, USA
- Ziegler J.F., Biersack J.P., and Littmark U. 1995. "The Stopping Range of Ions in Solids". Pergamon Press.

AUTHOR BIOGRAPHY

ROBERT WITTMANN was born in Vienna, Austria, in 1966. He worked from 1989–1997 as an engineer in the Development Department at the company Kap-sch, Vienna. After that he studied computer technology at the Technical University of Vienna, where he received the degree of 'Diplomingenieur' in 2002. He joined the Institute for Microelectronics in June 2002, where he is currently working on his doctoral degree. His interests include process simulation, Monte Carlo methods, and software technology.

ANDREAS HÖSSINGER was born in St. Pölten, Austria, in 1969. He studied technical physics at the Technical University of Vienna, where he received the degree of 'Diplomingenieur' in 1996. He joined the Institute for Microelectronics in 1996. In 1998 he held a visiting research position at Sony in Atsugi, Japan. In 2000 he finished his Ph.D. degree. He is currently enrolled as a post-doctoral researcher at the Institute for Microelectronics. In 2001 he held a position as visiting researcher at LSI Logic in Santa Clara, CA, USA within the scope of a cooperate research project on three-dimensional process simulation. His research interests include process simulation with special emphasis on three-dimensional applications.

SIEGFRIED SELBERHERR was born in Klosterneuburg, Austria, in 1955. He received the degree of 'Diplomingenieur' in electrical engineering and the doctoral degree in technical sciences from the Technical University of Vienna in 1978 and 1981, respectively. Dr. Selberherr has been holding the 'venia docendi' on 'Computer-Aided Design' since 1984. Since 1988 he has been the head of the Institute for Microelectronics and since 1999 he has been dean of the 'Fakultät für Elektrotechnik und Informationstechnik'. His current topics are modeling and simulation of problems for microelectronics engineering.

SIMULATION IN ENGINEERING

VIRTUAL PROTOTYPING OF A DOMESTIC ROBOT FOR DESIGN AND NAVIGATION OPTIMIZATION

Marcos Fernández
Sergio Casas
ARTEC Group, Valencia University
Edificio 2 Paterna, Polígono de la Coma
E-46980 Paterna (Valencia). Spain
E-mail: Marcos.Fernandez@uv.es

Ana Martínez
Laura Nuñez
Dorleta Guzmán
David Villaverde
Joseba Landaluze
IKERLAN Research Centre
E-20500 Arrasate-Mondragon. Spain
E-mail: aimartinez@ikerlan.es

KEYWORDS

HIL (Hardware in the Loop), Virtual Prototyping, Domestic robotics, Real Time simulation, Real Time graphics, Interactivity.

ABSTRACT

This paper describes the application developed as part of the “PROSIGRAT” research project carried out by IKERLAN and Valencia University’s ARTEC group dealing with the use of Real Time simulation for services and products evaluation. The application consists of the development of a Real Time graphic simulation of a domestic mobile robot moving in a virtual environment and guided by a real physical navigation and control system, based on the HIL methodology concept. This is a clear example in which the Real Time simulation of a mechatronic system interacting with the environment leads to a reduction in the time-to-market of a new product that has been well tested and operates in a safe, reliable way. The paper shows the details of the application and the main conclusions obtained.

INTRODUCTION

Nowadays competitiveness is a key point for the companies. This concept involves the reduction of costs and time-to-market when launching new products or new versions of existing ones. When it comes to the development of complex mechatronic systems including control elements, very often the integration of the control subsystems happens at a late stage of the product development cycle. The tuning of the control algorithms and the operating tests are carried out on a real prototype of a new product, in a process prone to error that could even cause a failure of the prototype.

That is why the use of new techniques that allow the validation of control systems from an early phase is so important. This is the main concept behind the application of the HIL (Hardware in the Loop) methodology. It consists of using realistic digital models of complex systems that interact with the real physical control systems (HW and SW) in the same way as the real complex system to be controlled. Another possible use of the same methodology

consists of validating several design alternatives designs for a new product in a realistic way by looking at different possible control alternatives.

A key point when developing HIL applications is the modelling of complex mechatronic systems in such a way that the model can be simulated afterwards in Real Time, and maintaining with high fidelity the evolution of the physical variables that are essential for the control system being developed. Although there are several commercial modelling and simulation tools oriented towards Real Time simulation of multidisciplinary systems, their suitability for any type of application is not at all clear (Otađuy et al. 2000).

The problem becomes even more complex when the physical system to be controlled interacts a lot with the surrounding environment. In this case the Real Time simulation of this environment and its evolution is also necessary.

This paper shows an example of the use of the HIL methodology explained above in order to aid the design of a new product: a domestic robot (vacuum cleaner application) and its control and navigation module. It is obvious that this is a case in which the system (robot) interacts with the environment (room to be cleaned) and, depending on the results of this interaction, the robot control system has to decide the new movement strategy to follow.

First of all the paper explains the main objectives of the research work carried out and the basic characteristics of the domestic robot application selected as a case study.

Secondly, a detailed description of the main application components is given. There follows a summary of the test and alternatives evaluation carried out with the help of the virtual prototype.

Finally, the main conclusions obtained from this work are presented, together with the future lines of research arising from this work.

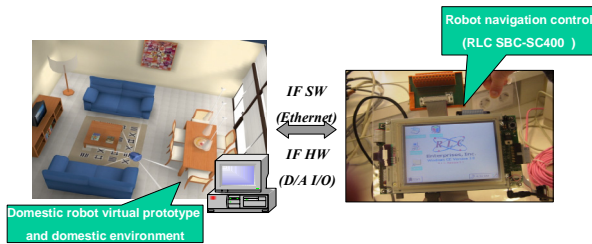


Figure 1: Robot virtual prototype for HIL

DOMESTIC ROBOT VIRTUAL PROTOTYPE

The main objective of the research work carried out by the Robotic Institute of Valencia University and IKERLAN as part of the PROSIGRAT project: “HIL based Virtual Prototyping and Real time Graphics Environment for Mechatronic systems” was to deal with the study and development of Real Time simulation and HIL techniques, as well as developments in the field of interactive 3D graphics. In order to assess the knowledge acquired during the project and the tools developed, as well as to validate the utility of the HIL techniques, the group planned to introduce some material objectives or demonstrators to the project. One of them was the Real Time simulation of a hydraulic cushion for a press in order to help the design and development of the control system, thereby avoiding having to use the real press (Landaluze et al. 2002a; Landaluze et al. 2002b).

The other demonstrator is the focus of this paper. It consists of the modelling and Real Time simulation of the components of an autonomous vacuum cleaner for domestic environments, as well as of the vacuum cleaner evolution environment, so that the navigation system could be developed and tested.

This application deals with a new product, a physical system that is still not available. So, many important decisions have to be taken regarding the best robot component design and configuration. The Real Time simulation of the robot moving in the domestic environment is also a way to aid this design.

A mobile robot is a complex system involving many multidisciplinary components: the actuators and drivers making up the locomotion system, sensors, energy supply systems, control SW and HW, etc.

The analysis carried out as part of the project focused on three components: the sensors to be installed in the robot to properly detect the environment, the locomotion system that improves the movements of the robot in a domestic environment, and the control strategies to ensure the robot moves in a safe and optimal way in different domestic environments (types of furniture, floor materials, types of soils, etc.).

As for the main elements (Figure 1) into which this domestic robot virtual prototype can be split, it is worth mentioning:

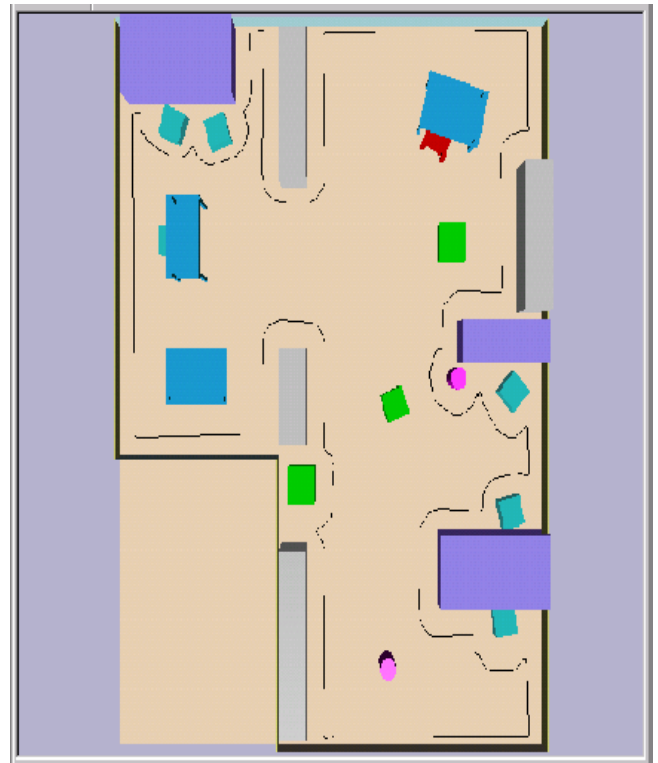


Figure 2: Wall following and obstacle avoidance

- The navigation system: the real physical part of the application consisting of a HW/SW platform similar to the one that would be integrated in a real domestic robot prototype.
- The simulation of robot dynamics, the graphic simulation of the room, and the simulation of the interaction between them.
- The interface between the real part of the prototype (control) and the simulated part.

There follows a detailed description of these parts of the prototype.

NAVIGATION SYSTEM

As previously explained this is the real physical part of the mobile domestic robot prototype. It consists of a group of SW modules running under the Windows CE operating system on a HW platform based on a Pentium 486 processor.

The navigation system consists of three basic stages: obtaining the contour to be swept, sweeping of inner contours and bordering of obstacles.

The first goal of the system is to obtain the outline of the robot's environment (Figure 2). For this purpose the robot uses a wall following strategy. A fuzzy control guides the robot along the walls. It takes the information provided by the ultrasonic sensors arranged all around the robot to gain a general perception of the immediate surroundings. Then, using this information and the robot's displacement direction vector, the speed and the steering angle required

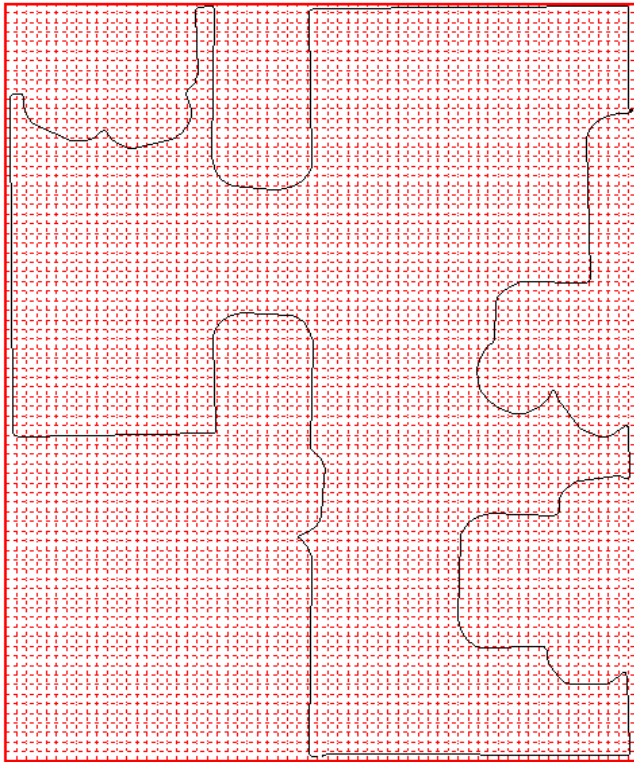


Figure 3: Area to be swept represented by a matrix

by the robot to follow the walls without colliding with any obstacle can be calculated (Urzelay et al. 1997).

It follows the walls until it reaches the starting point. This determines not only the contour of the initial navigation, but the area in which the robot must move. The contour will be limited by a rectangle whose area is as small as possible. This area is divided into cells of a prescribed size. Each cell has a status: the cell can belong to the contour, to the outside of the contour or to the inside of the contour. As a result the zone of intake will be limited to the inside of that contour.

Henceforth the behaviour of the prototype is not controlled by the fuzzy rules. At this moment the robot will use the matrix (Figure 3) obtained in the first round to calculate the points of the next path. Therefore some of the cells will change their status from “inside” to “path”. By connecting the cells that are in “path” status, a path with the same shape of the initial contour is obtained, but inside the previous one. The cells whose status was “path” are converted into “swept” cells. If the room was completely empty, the robot would sweep the room in a spiral.

But the behaviour is not completely planned. The robot is not only guided by this “map”, because there is the possibility of encountering an obstacle along the path. The robot goes around an obstacle in the same way it follows a wall, using the fuzzy control, until a point of the initially marked path is found. When the robot finds this point it is again controlled by the logged “map”.

The process of sweeping inner contours is repeated continuously until a 3% of the area initially selected for the task left. At that moment the prototype detects the zones

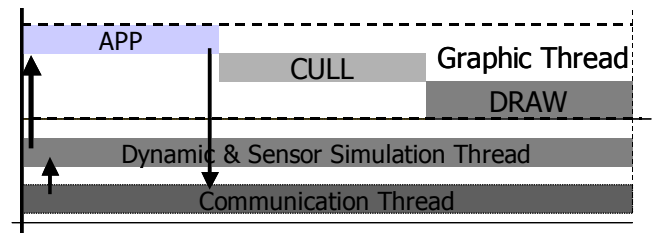


Figure 4: Multi-threading Scheme of graphic system

that have not been swept yet, called "islands". The robot only chooses the “islands” that are bigger than the size of the vacuum cleaner robot. Then the robot selects the “island” nearest to its actual position and moves ahead to it by the shortest route. Each island then becomes a “small room” to be cleaned, and the sweeping is controlled by the same rules of behaviour. Once all the islands have been swept the robot considers that the task is done.

To embed the navigation system in a Windows CE platform the fuzzy toolbox of MATLAB was first used to edit the fuzzy rules of control. Once the rules are known a model can be created in SIMULINK, including the fuzzy rules edited before in a block. Finally code can be generated with the RTW tool.

ROBOT DYNAMICS AND ENVIRONMENT GRAPHIC SIMULATION MODULE

As pointed out above, one of the key issues of the system presented is to provide a test environment for the HIL control strategies, which means being able to simulate the environment where the robot is working and to simulate the dynamic behaviour of the robot and its interaction with this environment, as well as the sensor information detected by robot sensors.

In order to have a better feedback and a visual first impression about control techniques, a Real Time 3D realistic simulation of the environment was required.

Apart from that, another requirement for the system was the flexibility to be reconfigured to simulate different HIL control and robot systems.

Module architecture

With the requirements indicated above, a module based on three main different subsystems was designed:

- 3D Real Time graphic subsystem.
- Dynamic simulation subsystem.
- Communication subsystem.

The three subsystems work simultaneously in a synchronized way to simulate the behaviour of the robot and its environment, receiving the control information from the HIL control system and sending back the information that the HIL control system will measure from environment sensors in a real system.

The system, which has Real Time requirements, runs in a multi-threading configuration. Figure 4 shows the three

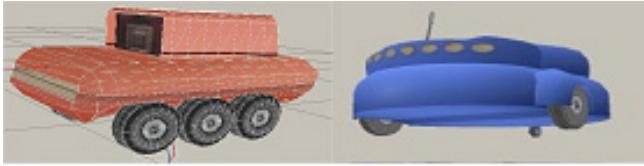


Figure 5: Example of library objects

main threads. Each of the threads is dedicated to running one of the three subsystems. The synchronization between subsystems is done in the application part of the graphic thread that is the base thread for this run-time environment simulation module.

The system configuration is based on the description of a correlated database that has two main components:

- Environment and robot graphic description.
- Dynamic robot and environment description.

Both descriptions are based on a hierarchical data structure represented in the form of two interrelated a-cyclic graphs. The interrelation allows the synchronized updating of visual and dynamic information, in this way maintaining the coherence between these two aspects of the system.

A brief description of the first two subsystems and their configuration now follows, whereas the communication subsystem is explained in the “Navigation-simulation interface” paragraph.

Real Time graphic subsystem

The Real Time graphic subsystem is based on a 3D graphics library call IRIS Performer (Rolf and Helman 1994). This library offers a run-time environment able to control a graphic application running at interactive frame-rates. For the system used a frame-rate of 30 frames per second was fixed, which was enough for the system requirements.

The basis for the 3D graphics system is the definition of a 3D graphic data structure. The data structure used was a *scene graph*. This scene graph is based on hierarchical a-cyclic graph, with different kind of nodes. These nodes allow the representation of geometry and visual properties (material, texture, lighting) of the environment elements, to configure the location and orientation of elements in the scene (Dynamic Coordinate System nodes) and also allow an optimisation of the time required to represent this scenario, which is a critical issue to maintain Real Time requirements in complex simulation scenarios. The optimisation techniques are based mainly on the level of detail management (LOD nodes) and culling of the scene parts not present in a particular frame (Funkouser 1993).

The image generation process is based on the traversal of this *scene graph*. In fact three different traversals are performed (corresponding to the App, Cull and Draw parts of the Graphic Thread represented in Figure 4). The first traversal is used to update the graphic data with the results obtained from the dynamic simulation of the system. Using this information the different mobile parts of the robots and

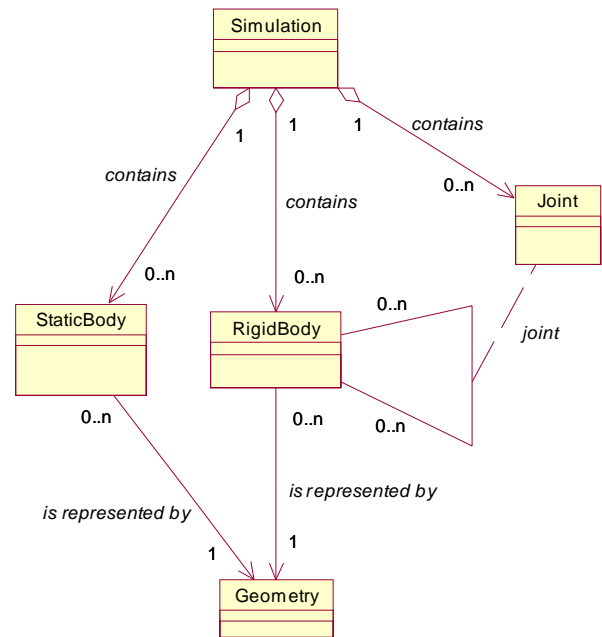


Figure 6: Simplified UML architecture

the scene are updated maintaining the consistency with the dynamic simulation graph. This traversal is also used for collision detection.

Once every object in the scenario has been updated the culling traversal provides the specific objects that have to be displayed in the following frame with its proper level of detail. The objects are sorted based on their material and texture in order to optimise the next traversal in the graphic thread. The last traversal is the drawing traversal, which is where the scene is actually rendered. The rendering is done using a 3D graphic pipeline schema.

In terms of the graphic part of the system a key issue is how to build the *scene graph* for a particular robot and its environment. To do so the system provides a graphic interface and the possibility of using a set of file formats that can be imported to represent the scenario geometries (as OBJ or VRML). It also offers a library of basic robot elements (wheels, robot bodies, etc., Figure 5) that allows the configuration of different environments in an easy way. It should be pointed out that the robot elements library not only provides the graphical scene graph to represent objects, but also includes its dynamic representation graph to simulate its behaviour.

Dynamic simulation

The aim of the dynamic simulation subsystem was to be a customisable and an easy-to-use modelling and simulation element. An element that allows different tests for physical systems to be built with no real difference between, for instance, a mobile robot and a hydraulic press, allowing at the same time a certain degree of specialization.

This came Newtonian dynamics, which are fairly computable, and realistic enough to get a mechanical simulation. Based on that idea, the concepts of rigid body and joints were introduced, due to the fact that most

mechanical systems can be modelled as several rigid bodies jointed together, with forces and torques acting over and among them. This can be accomplished by simulating a Newtonian dynamics system, combined with a Lagrange multiplier based constraint formulation in order to implement the joints.

At the same time, the concept of static body was introduced in order to simulate those objects that will never be moved under any circumstances. This is a way of simplifying things by letting the dynamics simulator only deal with those objects that are likely to move, and ignoring those that will never move. The representation of such a system is based on a hierarchical graph that builds a robot through the connection between these different elements. Apart from this, a correspondence with the graphic representation has to be established, which is accomplished by a set on links (represented as memory pointers) that allows navigation between graphical and mechanical data structures. In Figure 6 a representation of the relationship between these concepts is represented using UML notation.

A more specialized approach to the problem was needed. Although the main interest was in robotic simulation, more specialized robot concepts such as motors, wheels, encoders, and sensors were included as well. This flexibility allows generalization and functionality with such a combination making a powerful modelling and simulation tool for several purposes.

The dynamic simulation thread includes rigid bodies, static bodies and joints. These constraints are the key for building a system as complicated as required. In addition, this thread simulates object collisions and provides a friction model. All of these are made with the help of the dynamics ODE library (Open Dynamic Engine) (Smith 2000). This is the most CPU greedy thread, as it is intended to be executed at least 100-150 times per second.

Finally, as indicated, the system has to send feedback information to the HIL control system. This point is based on the simulation of different sensors that emulate the ones present in real robots. The sensors considered were as follows: contact sensors and ultrasound sensors. Contact sensors are simulated using collision detection algorithms between the robot body and the environment. Ultrasound sensors are simulated using ray-casting algorithms to simulate the radiation diagram of real sensors.

NAVIGATION-SIMULATION INTERFACE

In the description of the robot simulation environment it was indicated that a communication thread was also present in the architecture. This communication thread is in charge of sending the data from the HIL, updating the robot control and sending back the information obtained from sensors in the simulated environment.

Two interfaces are provided to implement the communication between navigation and simulation in the mobile robot simulation. The first one is a software socket communication and the second one is a hardware communication. The difference between them is that the

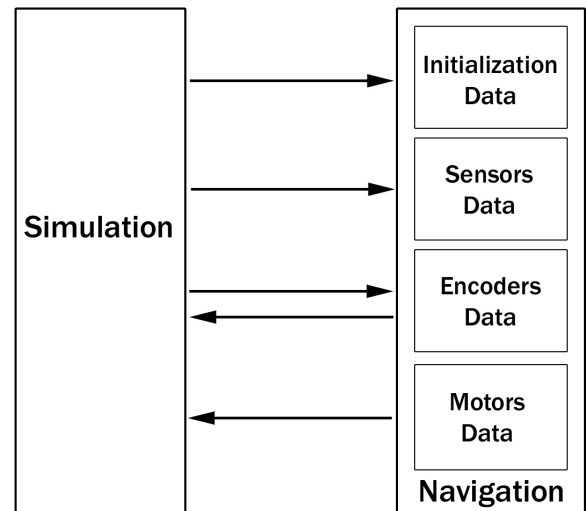


Figure 7: Simulation and navigation sockets

first establishes communication between the simulation software and the navigation simulation software, meanwhile the second establishes communication between the simulation software with the navigation hardware prototype.

As it is needed, the socket communication system is designed to send all the ultrasonic sensor and encoder measurements to the navigation system (from the simulation), and to send the desired motor speeds to the simulation (from the navigation).

In addition to this, there is a number of initial values to be interchanged, so an initial data communication is required.

With this fact in mind, it is not difficult to guess that four channels are provided to get and set (depending on the side considered) all the interchanging variables. These four channels are implemented as four sockets, one for each kind of communication. As the simulation acts as socket server, there are four client sockets (Figure 7) in the navigation system: *Initialisation Data*, *Sensors Data*, *Encoders Data* and *Motors Data*.

Initialisation Data is a channel for communicating general robot data (such as mass, lengths, etc), sensor positions, motor characteristics, and encoder parameters. This channel is also used for communication of additional data to the navigation for debugging purposes.

Sensors Data is a channel used to send (from simulation to navigation) the voltage representing the distances measured by the series of ultrasonic sensors fitted to the robot. This value is used by navigation to decide the next movement strategies.

Encoders Data is, as its name implies, the channel used to send (from simulation to navigation) the values measured from the robots's motor encoders. Encoders indicate the approximate angle the wheel has rotated, indicating if the wheel is rotating regardless of the order sent to the motor

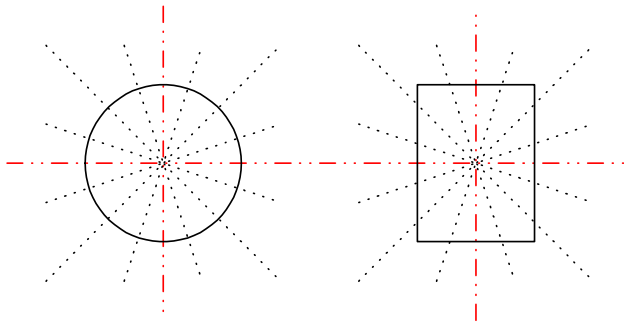


Figure 8: Sensor arrangement on robot

(which it may be impossible to get due to physical difficulties). This is a two-way channel because navigation can send some kind of orders to the encoder, such as a reset count signal.

Finally, *Motors Data* is the channel used to send (from navigation to simulation) the voltage representing the motor speed required. This communication establishes complete feedback between navigation and simulation, expressed as a loop in which the robot world and the sensors (encoders and ultrasonic sensors) are simulated. It sends sensor values to navigation, which decides what to do, sending motors values to react to the changes in the environment.

On the other hand there is the hardware communication system, which is in fact the real way to do the HIL design process.

The communication takes place using a ZWorld BL-2100 TCP-IP board system. This board takes an Ethernet input, which is output from the simulation PC, and has several digital and analogical output signals, which are inputs in the navigation hardware control system. This makes it possible to use in the simulation the same kind of socket structure mentioned before, with the only difference that the implementation of a program in the board ROM is needed in order to extract all the sockets data and convert them into voltage signals plugged into the navigation hardware.

EVALUATION OF PROJECT RESULTS

Evaluation of locomotion alternatives, sensorisation and navigation strategies

To validate the implemented navigation system, the simulator explained above was used. The simulator reproduces the virtual robot in addition to the environment which it operates in.

This simulation tool is flexible enough to permit the user to select the physical characteristics of the virtual robot. The user can select the shape and initial orientation of the robot, the configuration of the sensors (number, type, position and orientation) and the locomotion system (car-like, holonomic and wheel configuration).

The simulator is a tool that allows the real navigation control to be refined and tested, applying that navigation control to the initial configuration that the user selects for the robot.



Figure 9: Environment designed for the vacuum cleaner prototype evaluation

There follows a summary of the conclusions, based on the test results for different robot configurations.

The results depend to a large extent on the prototype shape. It is verified that for the same distribution of sensors and comparable volumes, a circular prototype moves through the room more continuously and its trajectory is smoother than the one of a rectangular shape.

Concerning the locomotion systems, the conclusion is that the car-like model does not satisfy the needs required for the final behaviour of the robot, as the turn radii were too large. To access specific zones, it was necessary to carry out complex manoeuvres that, as well as complicating the control, increased the time needed for the execution of the task. The selected system was finally the holonomic system and the rest of conclusions given here refer to a robot with that locomotion system.

A priori, it seems that due to the holonomic behaviour of the prototype, the most suitable sensor arrangement will be a symmetrical arrangement with respect to both axes (Figure 8).

The test results indicate that the prototype behaves better when its sensors are arranged in the radial direction. This was foreseeable due to the type of locomotion system chosen, as there is no privileged direction of movement. That is to say, what is the front part of the prototype at a given time, will probably not be shortly afterwards. This implies that the prototype should be able to receive the same quantity of information from the environment, whichever direction it is moving in. To obtain the desired result, the sensor arrangement must be symmetrical.

Whenever the symmetry of the sensor arrangement is maintained, its orientation does not necessarily have to be radial. However, the simulation demonstrated that the results are better with this radial configuration.

The minimum number of sensors that assures a correct behaviour of a 40 cm diameter prototype is twenty. Fewer



Figure 10: Active sensors in the vacuum cleaner robot

sensors increase the possibility of collisions with certain types of objects.

Initially, it was considered that the only condition the robot should verify was not coming into contact with objects in the environment. So only ultrasounds sensors were analysed. A subsequent analysis introduced the possibility of gentle robot contact, and the possible use of contact sensors.

Graphic and dynamic robot prototype configuration

To show the functionalities of the system a mobile vacuum cleaner robot prototype and its working environments were set up using the system.

The holonomic vacuum cleaner robot itself was modelled with four rigid bodies (a body framework and a locomotion system consisting of one free-rotation wheel and two drive and steering wheels), two motors (one traction motor for both wheels and one steering motor for the same two wheels) with its corresponding incremental encoders, and twenty four ultrasonic sensors arranged in a circle in order to measure distances.

The robot was treated as any other dynamic object in a customisable, changeable environment, in which its behaviour should be tested. The key to this kind of simulation is the fact that objects in the environment are of the same class as the robot itself, so an environment to test any system can be built easily. This is also the main problem, because any system can be built, whether it is physically consistent or not, and there could be non-stable systems and physical aberrations. But, anyway, this is actually a feature deriving from its structure, so it should not be considered as something the system lacks, but a consequence of its flexibility.

In this vacuum cleaner application, the dynamic simulation thread now includes the locomotion system (motor and wheel simulation with encoders to measure wheel rotations) and the sensor system, in addition to the previously mentioned rigid body constrained simulation world with friction and collisions.

The drawing thread remains the same, with the only exception of sensor drawing and colouring (Figure 9), which is an addition to offer a way to test the reliability of the sensors.

The communication thread is based on a two-way solution. The first one is a software communication system made with asynchronous communication sockets between the simulation and a navigation simulation. The second one is a hardware board communication system that communicates the mechanical simulation with the hardware navigation prototype.

To perform the navigation a scenario consisting of a living room and the furniture in it was prepared. In order to improve the appearance, radiosity was used to light the environment (Figure 10). Apart from the visual appearance, simulations were performed for different cases of environment properties that affect the dynamics (as friction coefficient of carpet and standard floor) and for different sensors (material with different coefficient of ultrasound reflection).

To run the system a 1.5 GHz Pentium 4, equipped with a 64 Mbytes GForce 4 MX440 graphic board was used. The system was able to run at 30 Hz. Figure 9 and Figure 10 show different aspects of the environment and robot set up for this prototype.

CONCLUSIONS AND FUTURE WORK

This paper describes the Virtual Prototyping of a Domestic Robot for HIL applications developed as part of the PROSIGRAT research project by IKERLAN and Valencia University.

The use of Real Time simulation of physical systems as a tool to help both in the design of new systems and in the developing of control elements has been proved to be really useful. However, it must also be mentioned that the development of reduced and reliable models of complex physical systems that reproduce their real behaviour is still a hard task that is far from being automated. This task is even harder when it comes to reproducing the interaction between a machine and its environment, as in the case of the example analysed in the paper.

As far as the graphic and simulation system is concerned, it is important to point out that it performed well for the prototype testing and evaluation. The system has also shown itself to be flexible enough to be reconfigured for other kinds of robots with different architecture and locomotion systems. In this way the simulation environment constitutes an interesting research and teaching tool because it allows real evaluations of control strategies in a simulated HIL environment, without the need for building the actual robots. This practice has been the first step in actual projects, where of course the final refinement has to be carried out with real robots. However, the system can save time during the control design process.

So the group working on this research project suggests that it is necessary to go on studying the establishment of a

working methodology to simplify at most the generation of complex systems models, oriented towards both Real Time simulation on low cost platforms, and the generation of interaction models between the system and its environment.

The current system performs well for controlled and low reactive environments. However, the simulation of complex reactive environments, where more complex control strategies need to be evaluated, will be considered in the future. The new kind of environments will include the simulation of force feedback from the environment, deformable and elastic materials and contact textures.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of the Spanish Ministry of Science and Technology for the development of this research as part of the PROSIGRAT project: "HIL based Virtual Prototyping and Real Time Graphics Environment for Mechatronic systems".

REFERENCES

- Funkouser, T.A. 1993. "Database and Display Algorithms for Interactive Visualization of Architectural Models". PhD. Thesis. University of Berkley.
- Landaluze J., I. Portilla, M.A. Otaduy, A. Martínez and R. Reyero. 2002a. "Virtual Prototyping of a Hydraulic Cushion for HIL". *ESM 2002*.
- Landaluze J., I. Portilla, A. Bidarte, A. Martínez and R. Reyero. 2002b. "Prototipaje Virtual de una Prensa Mecánica con Cojín Hidráulico". *XIV Congreso de Máquinas-Herramientas, Donostia*.
- Otaduy M.A., A. Martínez, A. Vidarte, J. Landaluze and R. Reyero. 2000. "Object-Oriented Modelling for Real-time Simulation". *2000 ASME (American Society of Mechanical Engineering) IMECE Automated Modelling Symposium*.
- Rohlf, J. and J. Helman. 1994. "IRIS Performer: A High Performance Multiprocessing Toolkit for Real-Time 3D Graphics". *SIGGRAPH Proceedigns*.
- Smith R. 2000. "Open Dynamic Engine". www.opende.sourceforge.net.
- Urzelay, J., J.P. Uribe and J.M. Ezkerra. 1997. "Fuzzy controller for wall following with a non-holonomous mobile robot". *Fuzzy IEEE*

TEACHING CONSTRUCTION STEPS BY USING VR

Gerardo Silva Chandía
Associate Professor
Department of Civil Engineering
U. of Santiago, Chile
Ecuador 3659 Office 1
Santiago, Chile
gsilva@lauca.usach.cl

KEYWORDS

VR, VE, engineering education.

ABSTRACT

Use of Virtual Reality (VR) applications in the construction industry in many countries is poor and many times inexistent by many reasons. One-way to introduce VR in this context in a definitively way, consist in the introduction of the VR tools in the undergraduate studies in the A/E/C (Architectural, Engineering and Construction) careers. In a “learning by doing” process, civil engineering (c.e.) students gain a direct appreciation of the advantages of use VR tools in the construction industry. As an academic experience, these tools are introduced to obtain a better knowledge retention as explained in the paper.

INTRODUCTION

One of the civil engineering (c.e.) teachers concerns is the persistence of the basic professional knowledge in the student’s mind after one academic period is finished.

In terms of Dr. Tulio Sulbaran (Sulbaran 2001): “The problem is the poor knowledge retention among engineering students, which is potentially related to the mismatch between student learning preferences and the media used to teach”.

Sulbaran y Baker (Sulbaran y Baker 2000) did find that “The (c.e.) students showed high levels of engagement and high levels of knowledge retention” by using a Distributed Virtual Reality (DVR) tool as support in engineering learning.

This paper author has find a positive students attitude in front of the Virtual Reality (VR) use in engineering teaching (Silva 2002, 2001, 2000).

The VR representation of building steps has been studied by Murray et al. (Murray et al. 2000) focusing in an eventually contractor/owner benefits.

They assume that “These models will then be used to represent the dynamics of the FutureHome construction site enhancing current construction practices that uses flat 2D plans and Gantt charts”.

Also the Computer Integrated Building Processes Group at NIST is doing research on the visual representation and simulation of construction and building related models, activities, and processes. The Virtual Reality Modelling Language (VRML) is being used to provide a 3D representation of those functions.

Their principal objective is the search and application of standard tools to stream the communications between the construction actors.

Another kind of research group works around Modelling and Simulation of construction processes using VR technology - 4D

A good idea of this approach is given in the VTT web page <http://cic.vtt.fi/4D/4d.htm>, site of the VTT Building and Transport firm.

We think the same as Aouad et al. (Aouad et al. 1999) have proposed, “The construction industry should also consider the rapid prototyping approach which can help solving most of the problems before construction begins. VR and 3D modelling tools provide the right medium for rapid prototyping purposes”.

TEACHING BUILDING STEPS

Teaching building steps or in a wide sense “construction steps” can be performed as any lecturer knows by many ways:

- Lecturer presents the theme and asks the students to read it in a textbook.
- Lecturer talks, students hear.
- Lecturer talks and writes in the whiteboard.
- Lecturer talks and “draw” in the w.b.
- Lecturer talks and presents a “slide show”.
- Lecturer talks first and continue with a presentation (animation, video, movie)
- Lecturer talks and introduces students with a VR tool that will support their own study.

This paper explores the assumed issues of the last listed teaching way.

In connection with the previous enumeration, it’s remarkable the “Cone of learning” presented by Dr. Nelson Baker in SUCCEED 2001 (Fig. 1)

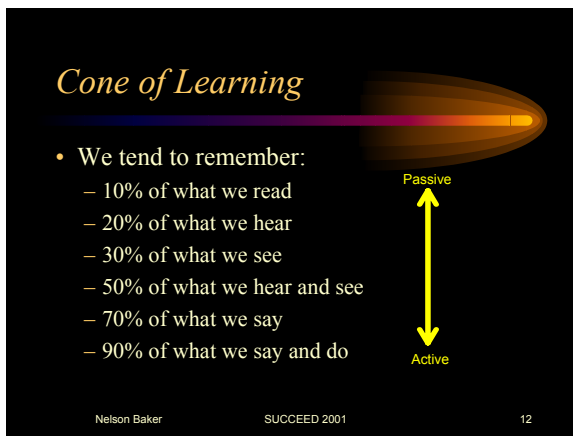


Figure 1: The Cone of Learning. After Baker, 2001.

PREVIOUS WORK

As previously published by this author, since 1998 civil engineering (c.e.) students at the U. of Santiago, Chile, they have created VR models by using 3D CAD (AutoCAD) first and then one or the two next options suggested to them:

- By using WalkThrough (WT) (AutoDesk) software (not supported since 1999) the students apply textures to the 3D model and they do freely fly through and walk through that can be recorded as animations in a proprietary format or in AVI format.

- By using 3D Studio to convert the 3D model into the VRML format.

3D models are published by the students in web pages containing:

- CAD file (DWG format or zipped)
- “Near VR model” (R13 DXF format for WT)
- VR model (VRML97 format)
- Video animations (streaming file format)
- Cost and time estimation and Gantt chart

Inspection of the contents of the student’s web page allows the teacher to grade them.

Inspection of the 3D CAD model and the 2D plans deduced from it, focus in the c.e. content.

Animations and the VRML file allows a quickly detection of mistakes or misunderstanding of the construction activities, not detected in the 3D CAD model.

Cost and time estimations are based on quantities calculated in a worksheet (Excel) based on textual and numerical attributes extracted from the 3D CAD model, constructed by using 3D “blocks” containing this kind of data.

TEACHING CONSTRUCTION STEPS BY USING VR

This paper author idea is similar to that presented by Marir et al. (Marir et al. 1998) in the VRML Application Section, as seen in the Figure 2.



FIG. 7: A VRML house model.

Figure 2: VR application, after Marir et al. (1998)

Methodology proposed by this paper author differs from the OSCONCAD as is explained below.

Proposed Teaching Methodology

This time the teacher does the 3D CAD model and the VR models.

The building steps are presented in two different ways:

- Animations (AVI format)
- “N” VRML files with N = number of construction activities considered

The teacher constructs the full 3D CAD model by using a clear layer structure.

This full model is the “N” building step.

By freezing the right layers each building step is generated.

Animations are generated by thawing the right layers, and looking at the 3D model from a fixed viewpoint or by doing a fly through or a walk through.

See Fig. 3 to 8.

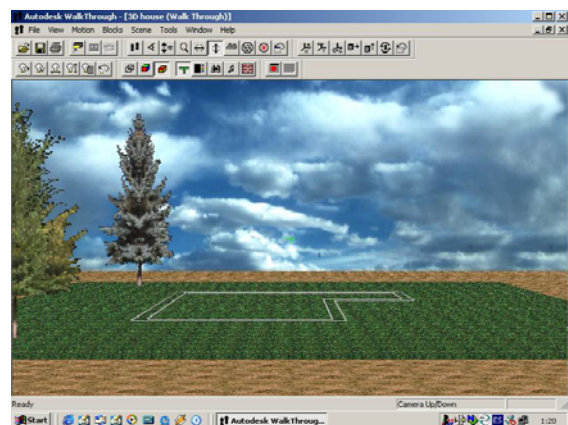


Figure 3: Tracer marks on site.

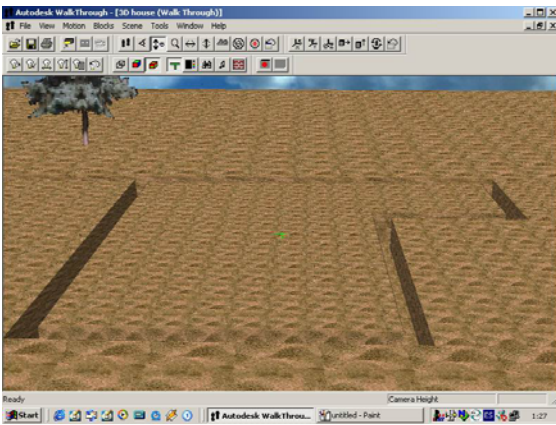


Figure 4: Excavations.

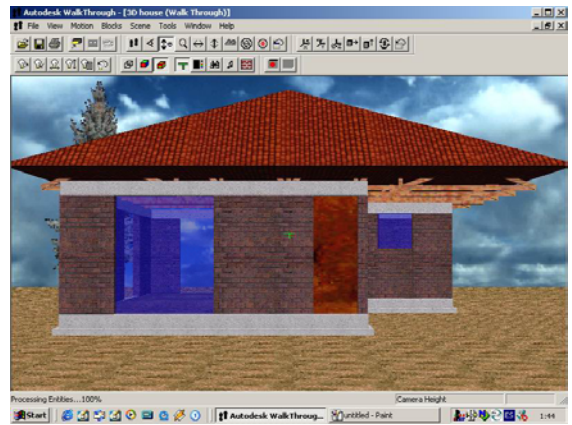


Figure 7: Roof, windows and doors.

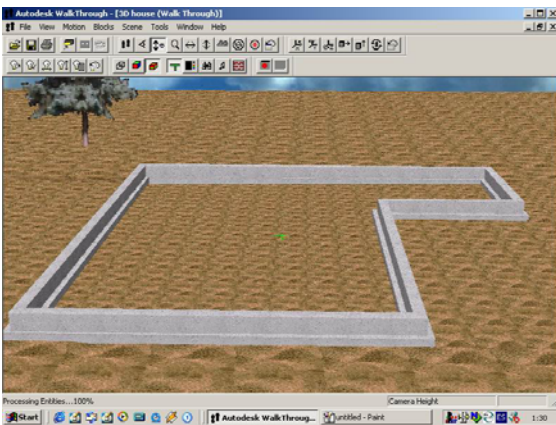


Figure 5: Foundations.

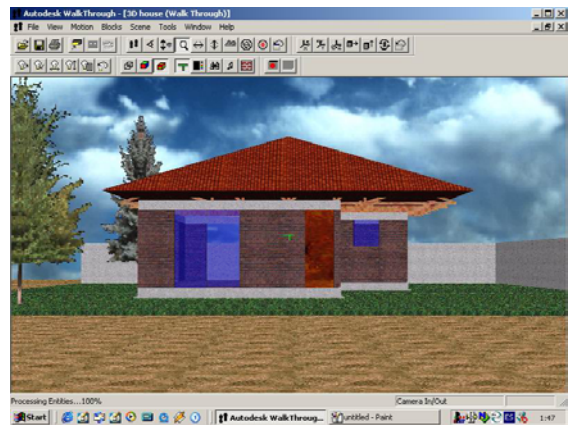


Figure 8: Finished 3D house.

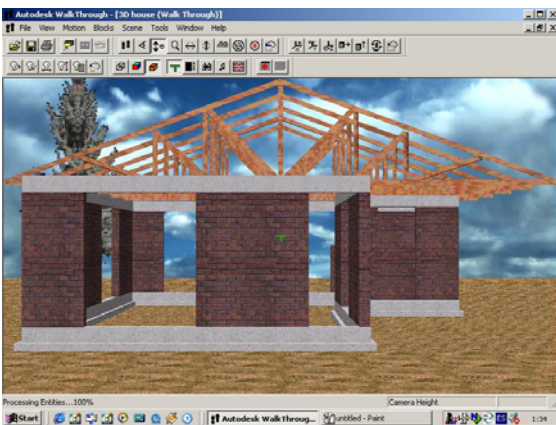


Figure 6: Walls, beams and roof structure.

VRML sequence of the building steps must be done manually by the students, inspecting the WRL files in the order of the list presented in the left frame of the browser (not shown this time by clarity). See Fig. 9 to 12.

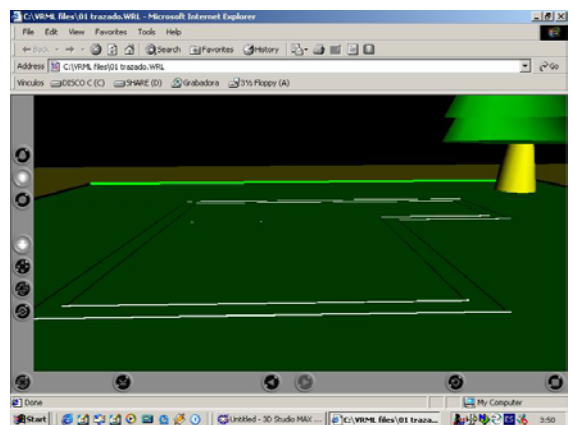


Figure 9: Tracer marks on site (VRML)

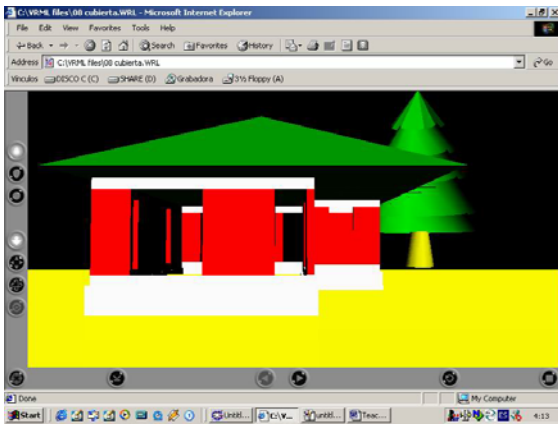


Figure 10: Building house to roof step (VRML)

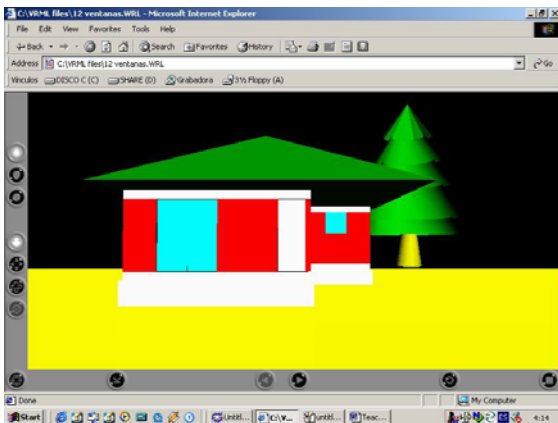


Figure 11: Interior finishes complete (VRML)

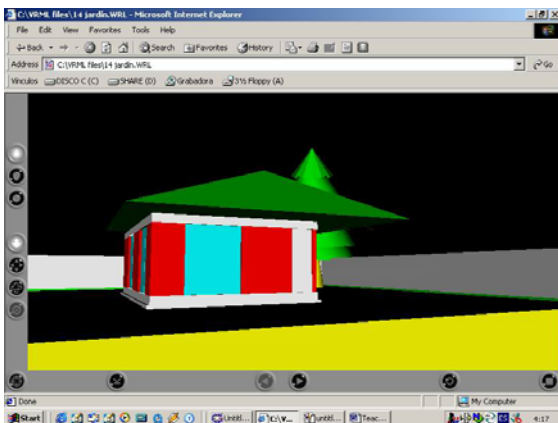


Figure 12: Exterior finishes complete (VRML)

A similar process can be done with other construction projects as bridges, pedestrian bridges, earth dams, dams, piers, industries, etc.

In many cases the first step is the modelling of the excavation in the natural site, which demands a great ability in the 3D model creation.

See Figures 13 to 15.

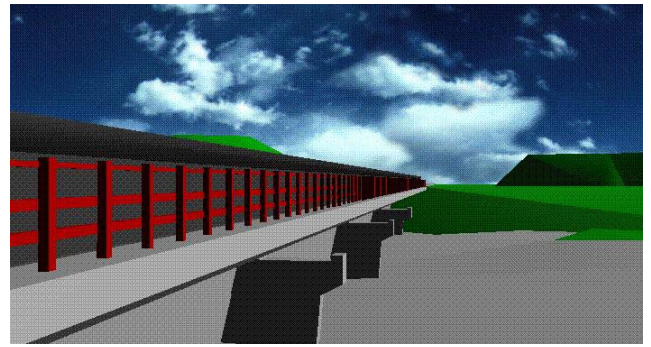


Figure 13: 3D model of a bridge.

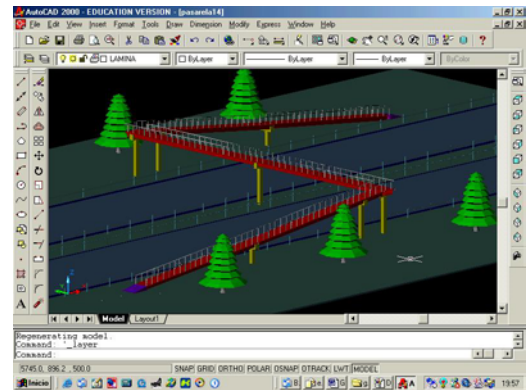


Figure 14: 3D CAD model of a pedestrian bridge.

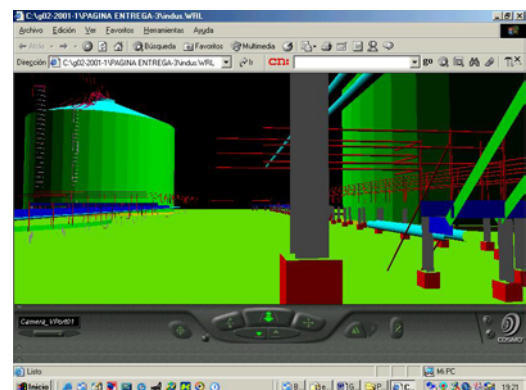


Figure 15: Industry 3D model (VRML)

CONSTRUCTION STEPS SIMULATION

Construction steps simulation is accomplished in this methodology, by using a kind of self-discovering in the students side.

In a web page, the student select a construction step description numbered in a list, and the VRML file is accessed, ready to be inspected by doing a fly by or a walk through.

The construction step is showed in the VRML file, complete, not in progress.

Construction machines and workers are not shown because the objective of the simulation is to present to the students the finished construction steps corresponding to construction activities such as they appear in a Gantt chart by example.

EXPERIENCE EVALUATION

This experience is planned to apply to students the first half of the 2003 year.

There will be two types of surveys:

- Students survey about the methodology
- Student's evaluation about construction steps
- Student's evaluation of knowledge retention after four months (academic period)

FUTURE WORK

Civil engineering students need to know much more than construction steps in a construction project.

They must know technical specifications, materials and cost estimations and time duration in each step.

This information can be presented in the same web page, in other frame, so the construction step presented in the VRML file can be interpreted in context.

More construction projects presented in this way, more knowledge will be trespassed to the students and they will gain a feeling about the range construction parameters will move in a particular case.

We will continue developing this new kind of didactic material, applying this methodology to others construction projects such as bridges, industrial buildings, etc.

Products will be published in:

<http://www.universidadesantiago.cl/dooce/vr>

CONCLUSIONS

Civil engineering students have now a new tool to support the learning construction steps process.

These tools are easy to use by teachers and students with CAD training.

So this kind of new didactic material can be developed with a time consuming as any other learning material.

The principal difference is the availability of this didactic material in the Internet WWW, which turns it in a free resource to support construction and building studies.

This work is possible to realize by using commercial software, easily available in the market, without computing complexities.

Senior c.e. students can easily complete this work by using commercial software to planning such as MS Project, because the 3D CAD models are built by using 3D "blocks" which contains attributes as element name, material, quality, with, length, height, etc.; making possible to realize any calculation in a simple spread sheet such as Excel.

REFERENCES

- Sulbaran T., "Impact of Distributed Virtual Reality on Engineering Knowledge Retention and Student Engagement", Georgia Institute of Technology, School of Civil and Environmental Engineering, Thesis Proposal, Summer 2001.
- Sulbaran T. and Baker N., "Enhancing Engineering Education Through Distributed Virtual Reality", FIE 2000, 30th ASEE/IEEE Frontiers in Education Conference, October 18 - 21, 2000 Kansas City, MO.

Silva G., "VR in Civil Engineering Education. Which is the Students Expectancy?", 6th International Conference on Information Visualisation, IV'02, London, England, July 10-12, 2002, Poster presentation.

Silva G., "VR and WEB Page Support in Civil Engineering Education: A Recent Experience", EUROMEDIA 2001, APTEC-1, UPV, Valencia, Spain, April 18-20, 2001.

Murray N., Fernando T., Aouad G., "A Virtual Environment for Building Construction", 17th ISARC, pp1137-1142, 18-20th September, 2000.

Marir, F., Aouad, G. and Cooper, G.S.; "OSCONCAD: A Model - Based CAD System Integrated With Computer Applications", ITcon Vol. 3 (1998).

AUTHOR BIOGRAPHY

GERARDO SILVA was born in Santiago, Chile and went to the Catholic University of Chile, at their Santiago site, where he studied civil engineering and obtained his degree in 1974. He worked for a couple of years for the same Catholic University as control laboratory engineer before moving in 1976 to the Civil Engineering Department at the University of Santiago, Chile, where he has been working in Soil Mechanics and Engineering Computing Applications up today.

COMPARISON OF OPEN-LOOP STIFFNESS CONTROL METHODS TO SUPPRESS SELF-EXCITATION VIBRATIONS

Kanjuro Makihara

Horst Ecker

Institute for Machine Dynamics and Measurement, Vienna University of Technology,
Wiedner Hauptstrasse 8-10/E303, A-1040 Vienna, Austria.

E-mail: kanjuro.makihara@tuwien.ac.at, horst.ecker@tuwien.ac.at

Tel: +43-1-58801-30312, Fax: +43-1-58801-30399

KEYWORDS

Vibration cancelling, Self-excitation, Open-loop control, Parametric excitation

ABSTRACT

Results of a stability investigation on vibration cancelling with three different types of variable-stiffness actuators are presented. A two-mass system is considered, with the top mass excited by self-excitation forces generated by a wind flow. The stiffness of an actuator between the base mass and the base changes, according to a control frequency. Comparisons of analytical predictions and numerical results are discussed. Numerical simulation is employed as a tool to investigate effectively the proposed methods and carry out parameter studies.

INTRODUCTION

Vibrations in mechanical systems can be generated intentionally or can be undesired, depending on the practical purpose of the system. If vibrations are not welcome, one has to find solutions either for avoiding the origin of the vibrations, or for cancelling or at least minimizing the vibration. In the case of linear time-independent systems a well-developed theory for system analysis is available and can be applied right away. However, many engineering systems exhibit nonlinear properties. Even if we restrict ourselves to mechanical systems, a wide range of nonlinear phenomena does exist.

As one of the approaches to prevent self-excited vibrations, a new idea of using parametric excitations to suppress self-excited vibrations was proposed (Tondl 1998). Within a certain frequency interval of the parameter excitation frequency, a phenomenon called *anti-resonance* occurs and the self-excited vibrations are fully cancelled in this interval. Another approach to prevent self-excited vibrations is an energy dissipative method. Among various variable-stiffness actuators, a nonlinear actuator was proposed in (Onoda et al. 1991). The actuator has an alternately attached and detached submember, leading

to the interesting characteristic that it provides no energy to the system. As the actuator has an inherent energy-dissipation mechanism, it can make the system more robust and needs only very little energy to be activated. For example, most space structures are far away from human maintenance. When there is a fault in the system, it can not be repaired easily. Therefore, reliability and simplicity are the main factors for the feasibility of such systems. The general feedback control requires a lot of equipment, such as sensors, processors and filters, which means that systems with complicated feedback control are more likely to fail.

In this paper, numerical stability investigations on vibration cancelling with variable-stiffness actuators that are activated by a certain control frequency are presented. The vibration control with open loop can be realized with nothing but actuators. It does not need to have sensors, filters or processors. Therefore the method can be applied to any machine with little equipment and preparation, which reduces the cost and makes the system more feasible and reliable.

MECHANICAL MODEL

We assume a two-mass system with self-excitation as shown in Fig.1, which is the same model as in (Ecker and Tondl 2000). The base mass m_2 is supported by a variable-stiffness actuator $k_{2-t}(t)$ and a damper with constant damping parameter b_2 . The actuator generates the force f_{act} according to the control laws. The top mass m_1 is attached to the base mass by a spring with constant stiffness k_1 and a constant damper b_1 . A constant flow U hits the top mass m_1 and causes a self-exciting force acting on the mass. The self-excitation is assumed to be of the Van der Pol-type.

$$F_{se} = b_0 U^2 (1 - \gamma_0 y_1^2) \dot{y}_1 \quad (1)$$

The equations of motion for the system are

$$m_1 \ddot{y}_1 + [b_1 - b_0 U^2 (1 - \gamma_0 y_1^2)] \dot{y}_1 + k_1 (y_1 - y_2) = 0, \quad (2)$$

$$m_2 \ddot{y}_2 + b_2 \dot{y}_2 - k_1 (y_1 - y_2) + f_{act} = 0. \quad (3)$$

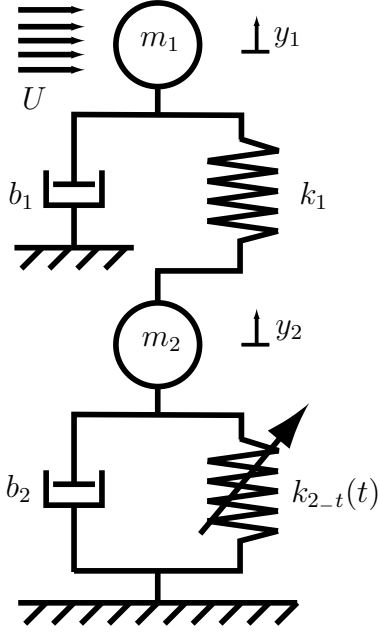


Figure 1: System with two masses, flow-generated excitation (U) and variable-stiffness $k_{2-t}(t)$.

MODELS OF ACTUATORS

We compare the following three kinds of actuators that are categorized as variable-stiffness actuators. Our control law is to adjust the stiffness of the actuators, according to functions with a control frequency ω .

Type 0: Harmonic parametric excitation

The variation of the stiffness of the actuator follows the harmonic function and the amplitude is determined by the parameter ϵ . This type of actuator has been applied intensively in (Tondl and Ecker 1999), (Ecker and Tondl 2000), (Ecker et al. 2002).

$$f_{act0} = k_{2-t}(t)y_2 = k_2(1 + \epsilon \cos(\omega t))y_2 \quad (4)$$

The coefficient of stiffness of the actuator changes harmonically with time as shown in Fig.2.

Type 1: Bang-Bang parametric excitation

This actuator provides two values of stiffness, according to the function with the control frequency ω . This system is frequently called Bang-Bang control. As described later, this function can be expanded to a number of harmonic functions, therefore it holds some potential for better performance of cancelling vibrations.

$$f_{act1-1} = k_2(1 - \epsilon)y_2, \quad \cos(\omega t) < 0 \quad (5)$$

$$f_{act1-2} = k_2(1 + \epsilon)y_2, \quad \cos(\omega t) > 0 \quad (6)$$

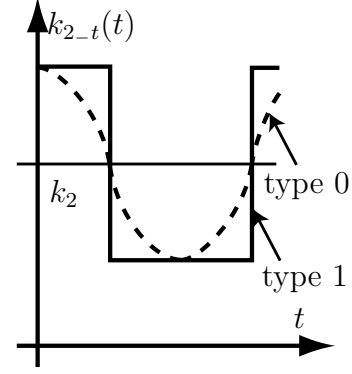


Figure 2: Variation of stiffness of type 0 and 1.

See Fig.2 for a graphical representation of this stiffness function.

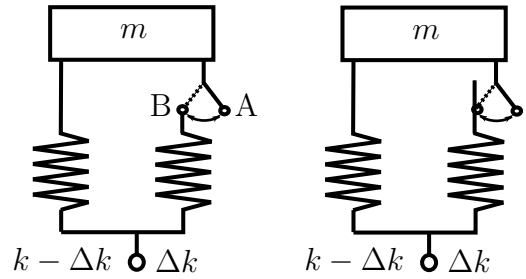


Figure 3: Actuator type 1 (left) and type 2 (right).

Type 2: Mechanical dissipative actuator

This actuator consists of two submembers. One is a spring with a spring parameter $k_2(1 - \epsilon)$ and the other is a variable spring that has a spring parameter $k_2\epsilon$, see (Onoda et al. 1991). For simplicity, in Fig.3 the constant spring is described as $k - \Delta k$ and the variable spring is described as Δk . The first spring $k - \Delta k$ is always connected to the base and the base mass. The second spring Δk is always connected to the base, but can be disconnected from the base mass. Since the disconnected spring of the actuator can be reconnected at any position, the neutral position may be shifted and the free length of the variable-stiffness actuator may change, as shown in Fig.3. The energy of the vibration can be transferred from low frequency modes to higher modes due to the stiffness variation of this actuator. As higher mode vibrations may usually be damped faster, damping of vibrations can be enhanced consequently. This actuator therefore provides a mechanical energy-dissipative mechanism.

Figure 4 schematically shows the load on the variable-stiffness actuator f against the displacement y . For a type 1 actuator, the state moves only along line $A'OA$ when the stiffness is high, and line OB

when the stiffness is low, respectively. The force jumps from point A to point B when the stiffness suddenly decreases and, inversely jumps from point B to point A as the stiffness suddenly increases. When the force jumps from point B to point A, the total energy of the system increases, which means that the type 1 actuator supplies some energy to the system. On the other hand, in the case of a type 2 actuator, the force similarly jumps from point A to point B in Fig.4 as the stiffness suddenly decreases, but remains at point B even when the stiffness suddenly increases. This time, when the stiffness increases, the state moves along line BC , whereas when the stiffness is still low, the state moves along line BO . The type 2 actuator can shift its neutral point (displacement of zero force) from $y_{np} = 0$ to $y_{np} = (\Delta k/k)y_0$. This is the essence of the difference between type 1 and 2 actuators.

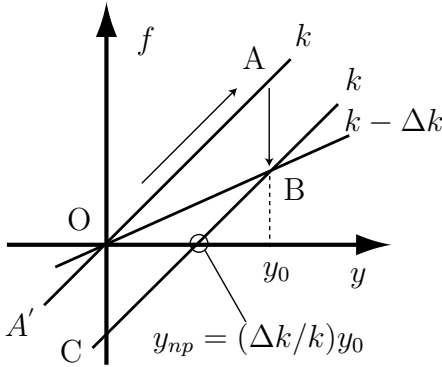


Figure 4: Characteristics of type 2 actuator.

When the variable submember is detached from the base mass, the force of the actuator is

$$f_{act2-1} = k_2(1 - \epsilon)y_2, \quad \cos(\omega t) < 0 \quad (7)$$

When the variable submember is attached to the base mass, the force of the actuator is

$$f_{act2-2} = k_2(1 + \epsilon)y_2 - 2k_2\epsilon y_0, \quad \cos(\omega t) > 0 \quad (8)$$

where y_0 is the displacement of the base mass just when the variable spring is reconnected to the base mass. This term of y_0 characterizes the system as a type 2 actuator.

Type 0 and 1 actuators induce parametric excitation and cancel the vibration of the self-excitation by the parametric excitation. It is obvious that a type 2 actuator primarily dissipates vibration energy. However, a type 2 actuator also induces parametric excitation in a more general way, and systems with such actuators are also classified as variable-stiffness system with open-loop control. Therefore it is possible to compare them from the viewpoint of stability and performance of cancelling

vibrations. For a fair comparison, we assume an average stiffness k_2 over one cycle of control oscillation and a bandwidth of the stiffness variation $2k_2\epsilon$.

TRANSFORMATION OF SYSTEM EQUATION

For a further analysis, it is convenient to introduce dimensionless coordinates and characteristic parameters. Non-dimensional displacements $x_j = y_j/y_{ref}$ can be defined with respect to a reference value y_{ref} .

$$\omega_1 = \sqrt{\frac{k_1}{m_1}}, \quad \tau = \omega_1 t, \quad x_j = \frac{y_j}{y_{ref}} \quad (9)$$

$$\eta = \frac{\omega}{\omega_1}, M = \frac{m_1}{m_2}, \kappa_1 = \frac{b_1}{m_1\omega_1}, \kappa_2 = \frac{b_2}{m_2\omega_1}, \quad (10)$$

$$\beta = \frac{b_0 U_0^2}{m_1 \omega_1^2}, V^2 = \frac{U^2}{U_0^2}, \gamma = \gamma_0 y_{ref}^2, q^2 = \frac{k_2}{m_2 \omega_1^2} \quad (11)$$

The equations of motion of the two-mass system are given in normalized form as

$$x_1'' + (x_1 - x_2) + [\kappa_1 - \beta V^2(1 - \gamma x_1^2)]x_1' = 0, \quad (12)$$

$$x_2'' - M(x_1 - x_2) + \kappa_2 x_2' + \hat{f}_{act} = 0, \quad (13)$$

where \hat{f}_{act} is defined as $f_{act}/m_2\omega_1^2 y_{ref}$.

Type 0: Harmonic parametric excitation

$$\hat{f}_{act0} = q^2(1 + \epsilon \cos(\eta\tau))x_2 \quad (14)$$

Type 1: Bang-Bang parametric excitation

$$\hat{f}_{act1-1} = q^2(1 - \epsilon)x_2, \quad \cos(\eta\tau) < 0 \quad (15)$$

$$\hat{f}_{act1-2} = q^2(1 + \epsilon)x_2, \quad \cos(\eta\tau) > 0 \quad (16)$$

Type 2: Mechanical dissipative actuator

$$\hat{f}_{act2-1} = q^2(1 - \epsilon)x_2, \quad \cos(\eta\tau) < 0 \quad (17)$$

$$\hat{f}_{act2-2} = q^2(1 + \epsilon)x_2 - 2q^2\epsilon x_0, \quad \cos(\eta\tau) > 0 \quad (18)$$

ANALYTICAL PREDICTIONS

Let us assume that the two-mass system has two natural frequencies, Ω_1, Ω_2 ($\Omega_1 < \Omega_2$). It is well known that the harmonic parametric excitation may destabilize the system in the vicinity of

$$\eta = \frac{(\Omega_j \pm \Omega_k)}{N} \quad (j, k = 1, 2; N = 1, 2, \dots). \quad (19)$$

However, the parametric excitation can stabilize the system in the presence of self-excitation. According

to previous works, stability of the trivial solution is possible for either $\eta = \Omega_2 + \Omega_1$ or $\eta = \Omega_2 - \Omega_1$ (Tondl and Ecker 1999), (Ecker and Tondl 2000). It depends on the situation if at all and at which control frequency η the stability of the system occurs. Previous studies tell us that the vibration can be cancelled by the harmonic parametric excitation with a type 0 actuator, if $\eta = \Omega_2 - \Omega_1$ is a so-called anti-resonance frequency. Let us define η_0 as $\Omega_2 - \Omega_1$.

The bang-bang function as shown in Fig.2 can be expanded into Fourier-series of harmonic functions with multiple periods of the original function.

$$\begin{aligned}\hat{f}_{act1}(\tau) &= q^2 \left[1 + \frac{4\epsilon}{\pi} \sum_{j=1}^{\infty} \frac{(-1)^{j-1}}{2j-1} \cos((2j-1)\eta\tau) \right] x_2 \\ &= q^2 \left[1 + \left(\frac{4\epsilon}{\pi} \cos(\eta\tau) - \frac{4\epsilon}{3\pi} \cos(3\eta\tau) + \dots \right) \right] x_2 \quad (20)\end{aligned}$$

The dominant trigonometric function of the series has the same period as the period of the bang-bang function, which means that the value of the dominant period is the same for the two functions. Therefore the system with a type 1 actuator is expected to behave in a similar way as the system with a type 0 actuator. From the above reasoning, we assume that $\eta = \Omega_2 + \Omega_1, 2\Omega_1, 2\Omega_2$ are resonance frequencies and $\eta = \Omega_2 - \Omega_1 = \eta_0$ is the anti-resonance frequency also for a type 1 actuator.

On the other hand, the type 2 actuator has an inherent energy-dissipative mechanism. Therefore, it is possible that a control frequency to achieve stability of the system with a type 2 actuator is different from the stabilizing control frequency of the system with parametric excitations. According to previous researches with type 2 actuators, a control frequency for stability is roughly $\eta \simeq 2\Omega_1$ (Onoda and Minesugi 1993, 1996). Actions of the attachment and detachment of the variable submember should be performed, near the time when the value of the product of displacement and velocity of structures is 0. However, the frequency of $\eta = 2\Omega_1$ is equal to a resonance frequency of parametric excitations, which means that the system excited with the frequency $2\Omega_1$ may be unstable. Therefore, the antagonism between this two opposite effects encountered in type 2 actuators is worth to be investigated further from the viewpoint of vibration cancelling. Details of the investigation of this interesting "stability competition" will be discussed in the next section.

NUMERICAL SIMULATIONS

For investigations of the dynamic behavior of the fully nonlinear system, the dimensionless equations were solved by means of numerical simulation to obtain maximum steady state amplitudes of

displacements. Three different software-packages (ACSL-Advanced Continuous Simulation Language, ACSL Math and MATLAB) were used for simulation and output representation. This effective combination of software provided a very fast, reliable and convenient environment to perform extensive parameter studies. An ACSL Math script controls all procedures and arranges parameter sets for the numerical simulation with the ACSL programs. While the calculation of ACSL proceeds, the ACSL Math script collects some values of variables from the ACSL simulation and creates an indicator for judging the degree of the stability of vibration. By using the indicator of the vibration, the ACSL Math script decides whether the vibration amplitudes have reached steady state conditions or not. When ACSL Math concludes that steady conditions prevail, the maximum amplitudes of displacement are retrieved from the time history. Otherwise, the ACSL simulation is prolonged for a more accurate simulation.

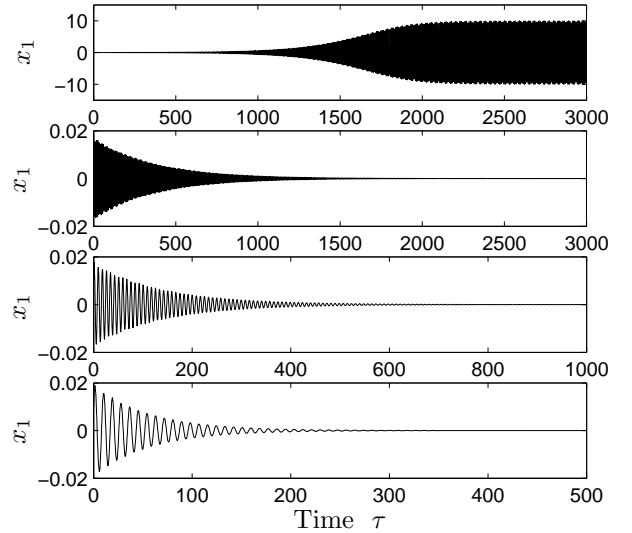


Figure 5: Time histories of amplitudes x_1 . Top to bottom: (a) without control, (b) type 0, (c) type 1, (d) type 2 actuator. See Table 1 for model data.

The parameter set shown in Table 1 was used for numerical simulations. In the case of $M = 4.0$, $\eta_0 = 2.47$, Figure 5 shows time histories of the displacement x_1 with the initial condition of $x_1 = x_2 = x_1' = x_2' = 0.01$. Figure 5(a) shows a time history without any control or with unsuitable control. The increasing vibration due to the self-excitation indicates that the uncontrolled system with self-excitation is inherently unstable, even though it has some damping elements. Figure 5(b) shows a time history with a type 0 actuator and vibration cancelling by the harmonic parametric excitation with the parameter $\eta = \eta_0$. Figure

Table 1: Parameter values for numerical examples.

M	var.	κ_1	0.03	q^2	6.0	κ_2	0.5
ϵ	0.2	β	0.06	γ	0.005	V	1.0
Ω_1	0.759	when $M=4.0$					
Ω_2	3.229	when $M=4.0$					
η_0	2.470	when $M=4.0$					

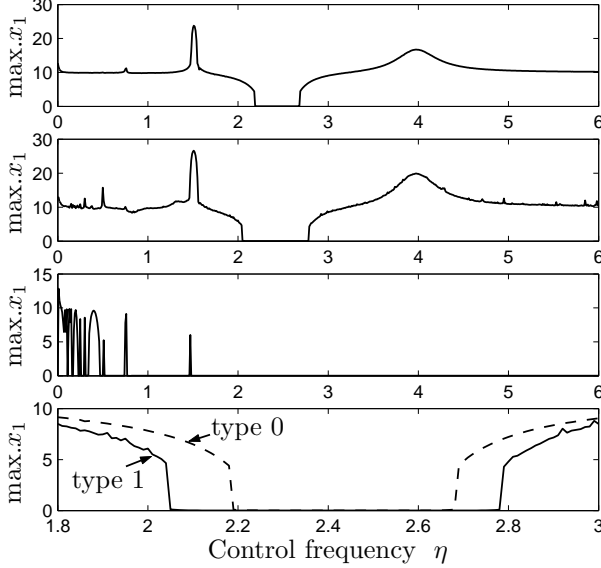


Figure 6: Range for maximum amplitudes x_{1_max} as a function of control frequency η with $M = 4.0$. Top to bottom: (a) type 0, (b) type 1, (c) type 2, (d) type 0 and 1 actuator.

5(c) shows a time history with a type 1 actuator. The vibration is cancelled by the bang-bang parametric excitation with the frequency $\eta = \eta_0$. Figures 5(b) and (c) prove that the idea of vibration cancellation by parametric excitations is possible. Note the different time scale of Figs.5(b) and (c), showing that the bang-bang control suppresses the vibrations significantly faster than the harmonic control. Figure 5(d) shows a time history with a type 2 actuator. Obviously the vibration is cancelled by the energy-dissipation mechanism with the parameter $\eta = \eta_0$. As long as we choose the correct parameters, each actuator can cancel the vibration caused by the self-excitation, otherwise the vibration increases.

Figure 6 shows the maximum amplitudes of the top mass x_1 after the system reaches steady state condition, as a function of the control frequency η and for the mass ratio $M = 4.0$. In Figs.6(a) and (b), plots for a frequency range of $\eta = 0.0$ to 6.0 show that the vibration of self-excitation was cancelled by the two types of parametric excitation with control frequencies near $\eta_0 = 2.47$. They also show two "peaks" of

resonance regions where the system is unstable by the parametric excitation, $\eta = 2\Omega_1 = 1.52, \Omega_2 + \Omega_1 = 3.99$. This result agrees with the analytical prediction from the last section. Fig.6(c) for a type 2 actuator shows that most frequencies keep the system stable unless the control frequency η is too low.

Near the frequency $\eta \simeq 2\Omega_1 = 1.52$ where the parametric excitation and the energy dissipation compete, the vibration was not cancelled. However, the maximum amplitude of x_1 with a type 2 actuator is smaller than those with type 0 and 1 actuators. It means that an energy dissipative mechanism suppresses the vibration to some degree at this control frequency. Furthermore, we also performed a numerical simulation with another set of parameters, which was expected to achieve "full quenching" in (Tondl and Ecker 1999). With this set of values, the vibration with a type 2 actuator was cancelled even when η is equal to $2\Omega_1$, whereas the vibration with the actuator 0 and 1 was not cancelled by the parametric excitation. Therefore, there is a need for further stability investigation with type 2 actuator and various parameter sets.

Figure 6(d) shows a comparison of the stable region of type 0 and 1 in zoom size. The anti-resonance frequency has some width, in the case of a type 0 actuator $\eta = 2.19$ to 2.68 and in the case of a type 1 actuator $\eta = 2.04$ to 2.78. The system with a type 1 actuator has a wider range of stable region than the one with a type 0 actuator. This result is quite reasonable when we keep in mind that the bang-bang control can perform more work on the system than the harmonic control, as can be seen from the stiffness function of the two control schemes in Fig.2. To put it in another way, the coefficient of a dominant trigonometric term of series expansion of the bang-bang function is $4/\pi$ times higher than the coefficient of the harmonic function in Eq.(20). The performance of vibration cancellation generally increases when the coefficient of the control amplitude increases. Therefore, the fact that type 1 actuators have a wider range of stable region than type 0 actuators was expected. Generally speaking, Dynamic Mass Dampers (also called Tuned Mass Dampers) have to be tuned very carefully because they work well only within a very narrow frequency band. In comparison, the great width of the anti-resonance frequency range of a parametric excitation is appealing and very significant when applied to actual systems.

Figure 7(a) shows the 3D-plot of the maximum amplitudes x_1 with a type 0 actuator, as a function of η and M . On this surface, a "canyon" is clearly visible representing the region of vibration cancelling by the parametric excitation. We see a steep "peak" in the region of $\eta = 1.0$ to 2.0, which represents the resonance frequency $\eta = 2\Omega_1$. We can also see a gentle

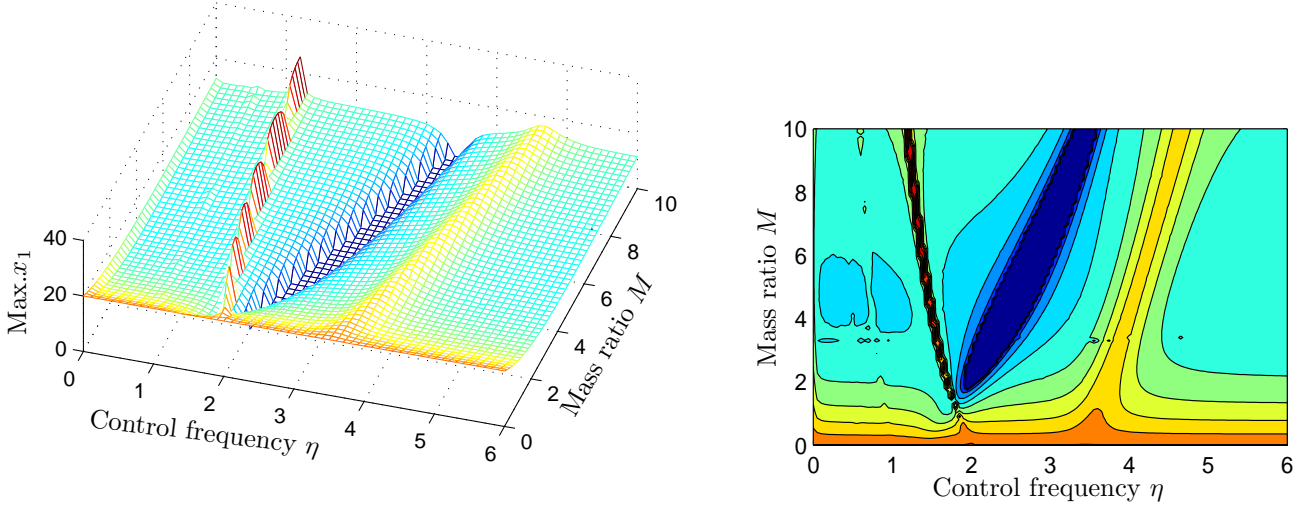


Figure 7: Maximum amplitudes x_{1_max} as a function of control frequency η and mass ratio M for a type 0 actuator. Left: (a) 3D-plot, right: (b) Contour-plot.

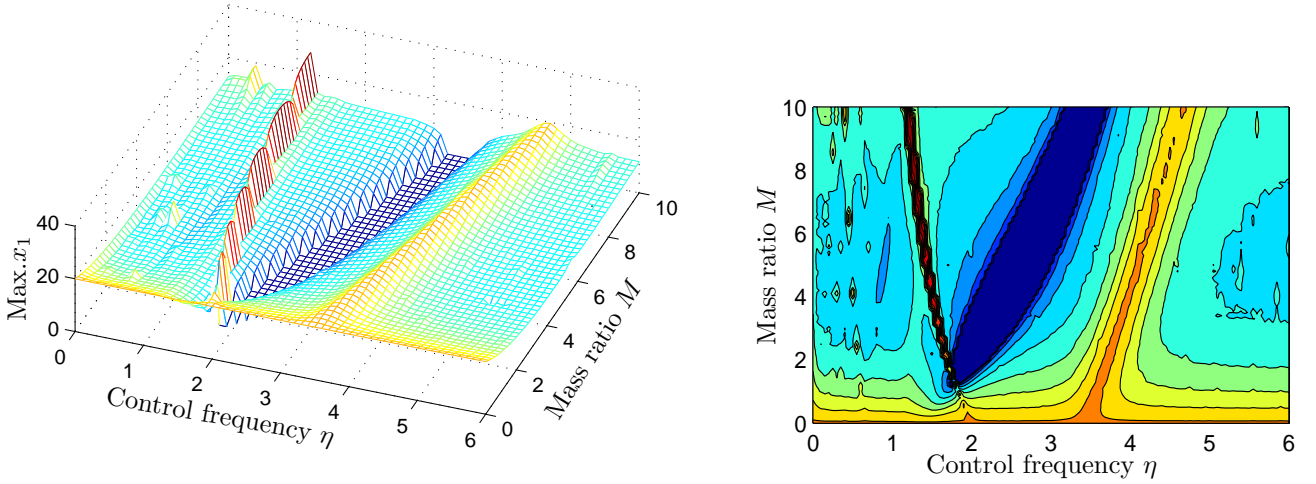


Figure 8: Maximum amplitudes x_{1_max} as a function of control frequency η and mass ratio M for a type 1 actuator. Left: (a) 3D-plot, right: (b) Contour-plot.

"peak" in the region of $\eta = 3.5$ to 4.5 , which represents the resonance frequency $\eta = \Omega_2 + \Omega_1$. Along these peaks, the system is parametrically unstable and the maximum amplitudes x_1 are larger than those without any control. The "canyon" and "peaks" have a curved shape, because the natural frequencies Ω_1, Ω_2 are a nonlinear function of the mass ratio M . Figure 7(b) shows the contour-plot of Fig.7(a).

Figure 8(a) shows the 3D-plot of the maximum amplitudes x_1 with a type 1 actuator, as a function of η and M . Here too, a "canyon" is clearly visible. Figure 8(b) shows the contour-plot of Fig.8(a). Comparing Figs.7(b) and 8(b), the system with a type 1 actuator has a wider area of stable region than the one with

a type 0 actuator. Note that the type 1 actuator is a bang-bang control and its stiffness changes sharply. Therefore the type 1 actuator also causes some other vibrations, not encountered with type 0 actuator and its harmonic and smooth stiffness variation. This explains the rugged surface in Fig.8(b) that can be seen especially in the low frequency range.

Figure 9 shows the 3D-plot of the maximum amplitudes x_1 with a type 2 actuator, as a function of η and M . The system with a type 2 actuator is stable for most combinations of η and M . The figure also shows two "peaks" of resonance regions where the system is unstable due to the parametric excitation, $\eta = \Omega_1, 2\Omega_1$. Therefore, the type 2 actuator

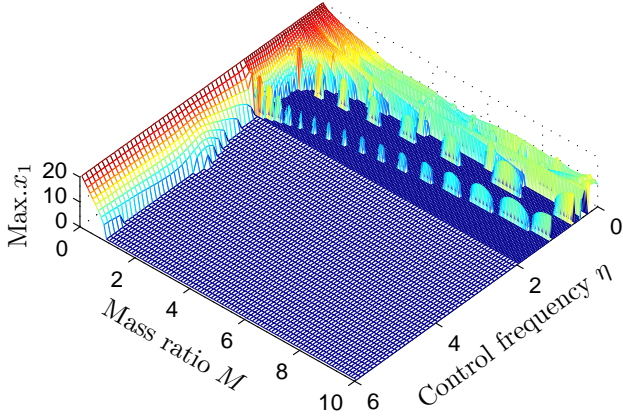


Figure 9: Maximum amplitudes x_{1_max} as a function of control frequency η and mass ratio M for type 2 actuator.

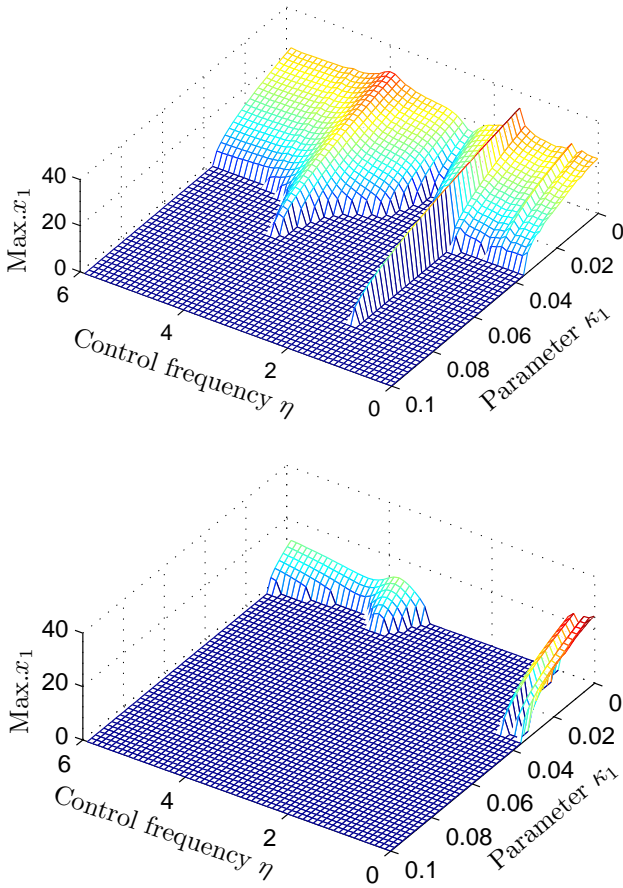


Figure 10: Maximum amplitudes x_{1_max} as a function of control frequency η and κ_1 . Top: (a) type 1, bottom: (b) type 2 actuator.

also shows the phenomena of parametric excitation to some degree. From the phenomena, it is possible that the system with a type 2 actuator also has an anti-resonance area, which is not visible on this chart, because the energy dissipative mechanism is dominant. A more sophisticated investigation is needed to evaluate the effect of each mechanism on vibration cancellation. In this case, this means that the type 2 actuator may be able to provide two kinds of mechanism of vibration cancelling at the same time. One of them is the mechanical energy-dissipation mechanism, and the other is the anti-resonance of the parametric excitation. This enlarges the possibility for vibration cancellation with type 2 actuators. The chart has discontinuous peaks at control frequencies $\eta = \Omega_1, 2\Omega_1$, because resonance peaks are very sharp and narrow. The simulation missed some peaks, due to the discretization procedure.

Figures 10(a) and (b) show 3D-plots of the maximum amplitudes x_1 as a function of η and the damping parameter κ_1 , as explained in Eq.(10), for type 1 and 2 actuators. Again we see the sharp resonance region where the system is unstable due to the parametric excitation frequency $\eta = 2\Omega_1$. Figure 10(b) shows that the system with a type 2 actuator also has some sharp unstable regions at small values of the control frequency η . κ_1 is a normalized parameter of the damping element b_1 in Fig.1. Even if the type 2 actuator provides no damping, the system is stable within control frequencies $\eta = 1.5$ to 3.2 .

Figures 11(a) and (b) show 3D-plots of the maximum amplitudes x_1 as a function of η and the damping parameter κ_2 for type 1 and type 2 actuators. The sharp resonance region where the system is unstable because of the parametric excitation frequency can be seen.

CONCLUSIONS

The stability of a system under the influence of a flow generated self-excitation with three open-loop variable-stiffness actuators was investigated. By using the combination of ACSL SIM, ACSL Math and MATLAB, we could perform extensive parameter studies on the stability in a fast and efficient way.

The type 1 actuator also has an anti-resonance frequency $\eta = \Omega_2 - \Omega_1$, which agrees with the analytical prediction. Although type 0 and 1 actuators behave in a similar way, the type 1 actuator has a wider range of the anti-resonance frequency for vibration cancellation by parametric excitation. The type 2 actuator excels both other types with respect to the stability range and shows great potential for practical applications.

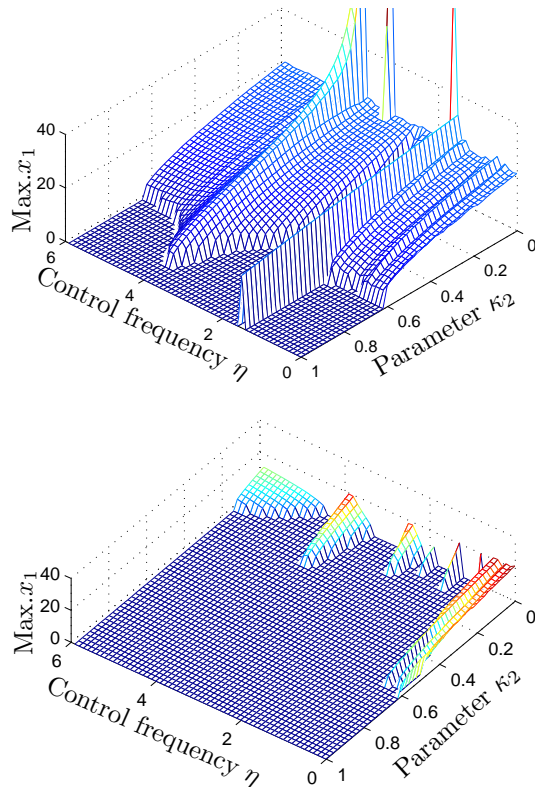


Figure 11: Maximum amplitudes x_{1_max} as a function of control frequency η and κ_2 . Top: (a) type 1, bottom: (b) type 2 actuator.

ACKNOWLEDGEMENTS

This contribution was partly funded by the Fonds zur Förderung der wissenschaftlichen Forschung (FWF) within project P16248-N02.

The first author appreciates the financial support by the scholarship of the Austrian Federal Ministry of Education, Science and Culture within a Joint Study Program between the University of Tokyo and the Vienna University of Technology.

REFERENCES

- Ecker H., Tondl A., 2000, "A Suppression of Flow-induced Vibrations by A Dynamic Absorber with Parametric Excitation", In Proc. of 7th Int. Conf. on Flow-Induced Vibrations, FIV2000, Lucerne, Switzerland.
- Ecker H., Pumhoessel T., Tondl A., 2002, "A Study on Parametric Excitation for Suppressing Self-excited Rotor Vibrations", Proc. of 6th Int. Conf. on Rotor Dynamics, Sydney, Australia.
- Onoda J., Endo T., Taoki H., N 1991, "Vibration Suppression by Variable-Stiffness Members", AIAA Journal, Vol. 29, No.6 (Jun), 977-983.
- Onoda J., Minesugi K., 1993, "Semi-Active Vibration Suppression of Space Structures", IAF Conference, Graz, Austria.
- Onoda,J., Minesugi,K., 1996, "Alternative Control Logic for Type-II Variable-Stiffness System", AIAA Journal, Vol.34, No.1 (Jan), 207-209.
- Tondl A., 1998, "To the problem of quenching self-excited vibrations", Acta Technica CSAV 43, pp. 109-116.
- Tondl A., Ecker H., 1999, "Cancelling of Self-excited Vibrations by Means of Parametric Excitation", ASME Design Engineering Technical Conference, Las Vegas, USA.
- The MATHWORKS, 2001, MATLAB Ver.6.1, Natick, USA.
- AEgis Tech. Group, 2003, ACSL-Advanced Continuous Simulation Language, Huntsville, AL, USA.

BIOGRAPHY



KANJURO MAKIHARA was born in Saga, Japan, and he is a doctoral candidate at the University of Tokyo, where he studies aerospace engineering. He performed analytical simulations with engineering models for solving impact problem of space debris, and obtained his bachelor's degree in 1998 and master's degree in 2000, from the University of Tokyo. He is currently working on research on innovative vibration controls of space structures based on energy-recycling methods for his Ph.D degree. From October 2002 to August 2003, he is at the Institute for Machine Dynamics and Measurement, Vienna University of Technology as a guest researcher and is working on the research of vibration cancelling in collaboration with Dr. Ecker.



HORST ECKER was born in Vienna, Austria, and graduated from Vienna University of Technology in Mechanical Engineering. His Ph.D. thesis was devoted to tire mechanics, but he soon discovered his interest in the application of numerical simulation in various engineering disciplines. As a result he coauthored on a book about ACSL in 1993. In 1995/96 he spent a year as a visiting scholar at Duke University, Durham, NC. Presently he is an Assoc. Prof. at the Vienna Univ. of Technology with a strong interest in nonlinear dynamics of mechanical and mechatronic systems.

Modeling of the Worm Gearing with Concave Shape Turn

Dr. Rasheed Abdulatif Abdullah
Al-Balqa Applied University
Tafilah Applied University College
P.O. Box: 183927 code 11118 – Amman- Jordan
Fax: +962-3-2250033
E-mail: rasheed_97@yahoo.com

KEYWORDS

Worm gearing, concave shape, model, grinding.

ABSTRACT

This paper presents the worm gearing with concave shape grind toroidal instrument –grinding wheel, whose cross section axis plane is arc of ellipse. The equations of surface grinding wheel were carried out. And the installation of this grinding wheel concerns to the grinding worm turns.

The worm gearing with concave shape turns (WGWCS) has a special place, in the worm gearing techniques. The difference from the other classical Worm Gears (archimedes worm, convolute worm, involute worm), this type of worm has in axis and cross section concave shape, but not Convex.

By its load – carrying capacity and the efficiency WGWCS are near to globidal. Which has by these parameters the superiority of the classical worm gearing with cylindrical worm. Therefore the WGWCS researching and developing is a very important scientific task. . The surface equation worm turn has been defined and carried out.

INTRODUCTION

The well -known WGWCS, the worm grind is a wheel grinding, its cross section axis plane is a circle arc (Daizhong and Hongye.,1986, Airapetov et al.1996).

In WGWCS, the worm grind is a wheel grinding its cross section axis plane is an circle arc with radius ρ . The radius ρ is chosen equal the reference worm radius, the angle γ_l crossing the worm axis the instrument are chosen the helix axis vaunt line on cylinder radius ρ .

The disadvantages of the WGWCS are (Niemann and Heyer 1953):

- The diameter decreasing of the grinding wheel in consequence regrinding cause decreasing the distance between the instrument axis and the worm. Which cases the distortion on the grinding worm surface.

- The contact line form in the worm gearing which make the grip danger cant be eliminate.

The contact line instrument surface and the worm (Illes1983, A.C 139531 1972), can not be spatial, but a flat arc which coincide with profile

instrument. This can be achieved by calculating the instrument parameters. In this case, the worm surface is determinated by not difficult equations.

WORM GEARING MODELING

On the ellipse surface by coincidence its centers to coordinate center, can be defined:

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} = 1 \quad (1)$$

where a - large major axis which lay out by X axis. b - short which lay out by Y axis. When b value proximity to a value, $\frac{b}{a} \approx 1$. So the

equation (1) will approximate to the circle equation. If $b = a$ so (1) will be

$$\frac{X^2}{a^2} + \frac{Y^2}{a^2} = 1 \quad (2)$$

The circle radius a will be in the center of coordinate. Thus, cross section axis plane in arc form of ellipse is near enough to axis plane in arc form of circle.

Figure 1 presents cross section axis surface wheel grinding in arc form of ellipse.

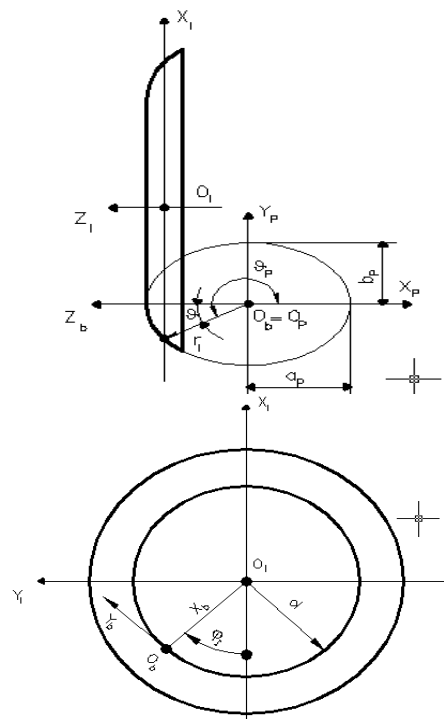


Figure 1 section axis surface wheel grinding.

Figure 2 presents worm and instrument vertical flat passed the middle point M worm profile. Assume that the line between the worm axis and the instrument A_I passing the M worm profile. In this point the radius vector length, which define its state on ellipse arc equal reference radius worm r_0 . Connecting the coordinate system to the grinding wheel O_I , as it shown in fig (1).

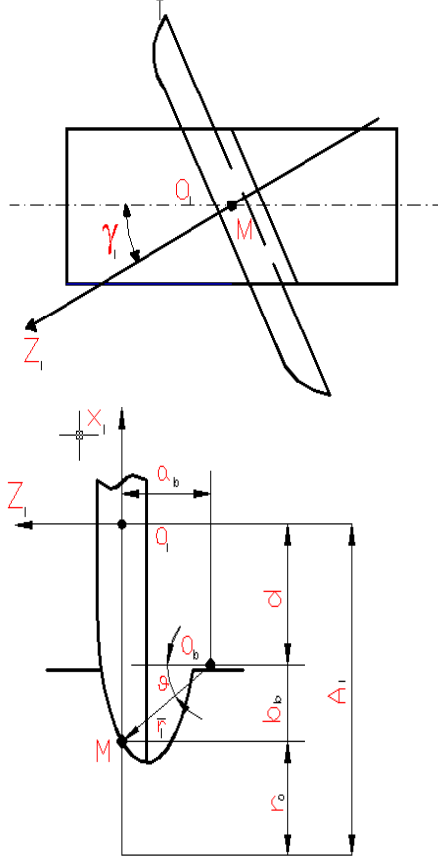


Figure 2: Worm and instrument

The coordinate system O_b is an auxiliary one in O_p coordinate system described by ellipse. The initial of the two systems is coincided. The column radius vector defined optional ellipse point in coordinate system O_p , it is given in the form:

$$\bar{r}_p = \begin{bmatrix} a_p \cos \mathcal{G}_p \\ b_p \sin \mathcal{G}_p \\ 0 \\ 1 \end{bmatrix} \quad (3)$$

where a_p - large major axis of ellipse which lay out by X_p axis; b_p - small major axis of ellipse which lay out by Y_p axis.

\mathcal{G}_p - angel between positive direction X_p axis and vector radius, defined the given point ellipse, measured anti- clock wise.

For the middle point M worm profile, in which as we said earlier, the radius length vector, define the point place on the ellipse arc, length equal radius worm r_0 can be written:

$$(a_p^2 \cos^2 \mathcal{G}_{pM}) + (b_p^2 \sin^2 \mathcal{G}_{pM}) = r_0^2 \quad (4)$$

where \mathcal{G}_{pM} - the angle \mathcal{G}_p in the middle point M worm profile.

Using the necessary mathematical transfers the (4) equation we get :

$$\operatorname{tg}^2 \mathcal{G}_{pM} = \frac{a_p^2 - r_0^2}{r_0^2 - b_p^2} \quad (5)$$

let us take a special case when $\mathcal{G}_{pM} = 225^\circ$.

Under this condition the equation 5 will be:

$$\frac{a_p^2 - r_0^2}{r_0^2 - b_p^2} = 1 \quad (6)$$

After some necessary mathematical transfers, taking into account that $b_p > 0$ the equation 6 will be:

$$b_p = \sqrt{2r_0^2 - a_p^2} \quad (7)$$

from (7) it become clear that $a_0 < r_0 \sqrt{2}$.

As it well-known, the ellipse eccentricity ε is between 0 and 1. Taking into account equation (7) so it can be as:

$$\varepsilon = \frac{\sqrt{a_p^2 - b_p^2}}{a_p} = \frac{\sqrt{2(a_p^2 - r_0^2)}}{a_p} \quad (8)$$

from (8) its become clear that $a_p > r_0$.

As follows from the equations (7) and (8), that while choosing the a_p value it is necessary to take into account the conditions :

$$r_0 < a_p < r_0\sqrt{2} \quad (9)$$

In figure (3) it is shown the graphical dependence of the short poles ellipses length b_p . On the long poles a_p in $r_0 = 46$ mm and, $\vartheta_p = 225^\circ$. The figure 4 shows the dependence ellipse eccentricity ε_p on length its long poles in the same conditions.

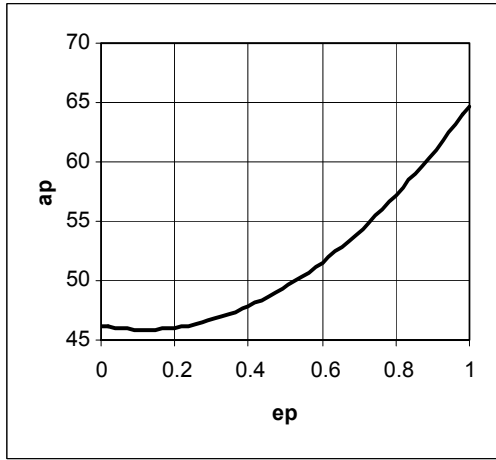


Fig 4: The ellipse eccentricity dependence on the length .

Analyzing figure 3 and figure 4, we realize that, choosing the a_p value corresponding to (9) equation has continuous series of ellipses, which have eccentricity in all the directions between 0 and 1.

For more comfortable calculations we can use another equation with the same purposes:

$$\vartheta = \vartheta_p - 180 \quad (10)$$

The angle γ_I crossing the worm axis and the instrument (see fig 2). It must be chosen the equal to slop angle screw line on the cylinder radius r_0 .

In the coordinate system O_I the center state symmetrical ellipse is given by a_b and b_b belongs to point M worm profile.

$$a_b = r_0 \cos \vartheta \quad (11)$$

$$b_b = r_0 \sin \vartheta \quad (12)$$

Radius grinding wheel equal the sum values of b_b and d - the distance from the coordinate center O_I to O_b measuring by X_I axis (fig. 2).

The center distance A_I connected by the relation:

$$A_I = r + d + b_b \quad (13)$$

Surface grinding wheel will form rotational arc profile (3), whose situation can be defined from the equations (10), (11) and (12), around the grinding wheel axis.

In this case:

$$\bar{r}_I = M_{Ip} \bar{r}_p \quad (14)$$

where \bar{r}_I - column radius vector point profile in coordinate system O_I ;

M_{Ip} - matrix transformation from coordinate system O_p to coordinate system O_I .

Matrix transformation from coordinate system O_p to coordinate system O_I can be defined as follows:

$$M_{Ip} = M_{Ib} M_{bp} \quad (15)$$

where M_{bp} - transformation matrix from coordinate system O_p to coordinate system O_b ;

M_{Ib} - transformation matrix from coordinate system O_b to coordinate system O_I .

The transformation matrix from coordinate system O_p to O_b correspond fig.1 defined as follows:

$$M_{bp} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (16)$$

The transformation matrix from coordinate system O_b to O_I correspond fig.1 defined as follows:

$$M_{Ib} = \begin{bmatrix} \cos \varphi_I & \sin \varphi_I & 0 & -d \cos \varphi_I \\ -\sin \varphi_I & \cos \varphi_I & 0 & d \sin \varphi_I \\ 0 & 0 & 1 & -a_b \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (17)$$

where φ_I the turning angle coordinate system O_b according O_I fig1.

Thus, taking into account the equations (15), (16) and (17), the transformation matrix from coordinate system O_b to coordinate system O_I will be :

$$M_{Ip} = \begin{bmatrix} 0 & \cos\varphi_I & -\sin\varphi_I & -d\cos\varphi_I \\ 0 & -\sin\varphi_I & \cos\varphi_I & d\sin\varphi_I \\ -1 & 0 & 1 & -a_b \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

Follows the surface of grinding wheel on the base of (3), (14) and (18) defined equation:

$$\bar{r}_I = \begin{bmatrix} \cos\varphi_I (b_p \sin\vartheta_p - d) \\ \sin\varphi_I (d - b_p \sin\vartheta_p) \\ -(a_b - a_p \cos\vartheta_p) \\ 1 \end{bmatrix}. \quad (19)$$

Using equation (19) we can find out the surface equation worm turn

CONCLUSIONS:

The result of the analysis was carried out, as:

1. Cross section action grinding wheel (in circle form) for the worm gearing with concave shape can be modeled and simulated by ellipse arc and corresponding equation (3) -(13).
2. When $\vartheta_p = 225^\circ$ has a continuous series ellipses, has eccentricity in all its defined field (from 0 to 1).
3. The surface equation worm turn has been carried out (see equation (19))

REFERENCE

- A.C 139531 (USSR) 1972 "Special form Cylindrical Worm Gearing"-M. -L.: Mach,
- Airapetov E., D Su and J B Hull, 1996, 'Analysis of the length of contact lines of worm Gearing', proceedings, the International Conference on the Theory and Practice of Gearing, 4-6 December, Izhevsk, Russia, pp 337-342.
- Daizhong S.U. & Hongye W.U., 1986 "Numerical Analysis Method of the Tooth Surface Load Carrying Capacity of Cylindrical Worm Gearing", Proceedings, National Conference On the New Technique of Machine Design, May 1986, Beijing, China, Vol. 8, paper No. 117.

Illes Dudas. 1983 Verfaurensmethoden zur Berechnung und leerstellung von Hohlphan.- Magdeburg,

Niemann G., Heyer E. 1953 Untersuchungen an Schneckengetrieben, VDI, Heft 6, , page 147-157.

Biography

Dr. Rasheed Abdulatif Abdullah: was born in Amman, Jordan, in 1968. He received the M.Sc. degree in Mechanical engineering from the Kiravograd Institute (Ukraine) in 1992. And the Ph.D. from the Kiravograd Technical University in 1998. Sense 2002 he joined Al-Balqa Applied University(Tafila Applied University College), Jordan, where he is currently a lecturer at the Mechanical Engineering Department.

COMPUTER SIMULATION OF COMPLEX SHIP SYSTEM GAS TURBINE – SYNCHRONOUS GENERATOR

Ante Munitic

Marijo Orsulic

Josko Dvornik

Split College of Maritime Studies

Zrinsko-Frankopanska 38, 21000 Split, Croatia

e-mail: munitic@pfst.hr, josko@pfst.hr

KEYWORDS

System dynamics, modelling, synchronous generator, twin shaft gas turbine, continuous and discrete simulation.

ABSTRACT:

System Dynamic Simulating Modelling is one of the most appropriate and successful scientific dynamics modelling methods of the complex, non - linear, natural, technical and organisational systems. The methodology of this method, together with use of digital computer, showed its efficiency in practice as very suitable means for solving the problems of management, of behaviour, of sensibility, of flexibility of behaviour dynamics of very complex systems. All this is made by computer simulating, i.e. “in laboratory”, which mean without any danger for observed realities.

The aim of this paper is to show the efficiency of the application of the System Dynamics Simulation Modelling in investigation of behaviour dynamics, one of the complex marine system and process i.e. “gas – turbine generator”. Twin shaft gas turbine and synchronous generator without contact shall be presented with mental - verbal, structural and mathematical - computing modules, and will simulate working process of ship propulsion complex.

1. SYSTEM DYNAMICS SIMULATING MODELLING OF TWIN SHAFT GAS TURBINE

Basic equations of twin shaft gas turbine:

Most often dynamical analyze of gas turbine is based on observation of plan as accumulator of kinetic energy, while dynamics of thermal energy can be conditional ignored.

We presume that process of fuel combustion is momentarily, condition of pressure in turbine is constantly, engine air rate is equal as gas consumption, and parameters of atmospheric pressure are relatively unchanged and ideal heat transfers. Linear behaviour of plan is available when fuel consumption not depends on angular velocity of turbo compressor rotor.

Equations of shaft of turbo compressor with low pressure – consumer:

$$T_{a1}\varphi_{\omega1} + k_1\varphi_{\omega1} = \mu_G + k_{\omega2}\varphi_{\omega2} \quad (1)$$

$$\frac{d\varphi_{\omega1}}{dt} = \frac{k_1}{T_{a1}} \left[\frac{\mu_G}{k_1} + \frac{k_{\omega2}\varphi_{\omega2}}{k_1} - \frac{f(t)}{k_1} - \varphi_{\omega1} \right] \quad (2)$$

$$\frac{d\varphi_{\omega1}}{dt} = \frac{1}{T_{a1}} (\mu_G + k_{\omega2}\varphi_{\omega2} - f(t) - k_1\varphi_{\omega1}) \quad (3)$$

Mental- verbal model:

When relative variation of fuel consumption μ_G and product $k_2*\varphi_{\omega2}$ are increasing, relative angular speed variation is increasing also, resulting in positive cause- consequence relation UPV (+).

When relative variation of possible external act of cargo and product $k_1*\varphi_{\omega1}$ are increasing, relative angular speed variation is decreasing and observed UPV (-) is negative.

Further, when time of shaft running T_{a1} is increasing, relative angular speed variation is decreasing, resulting in negative sign UPV (-).

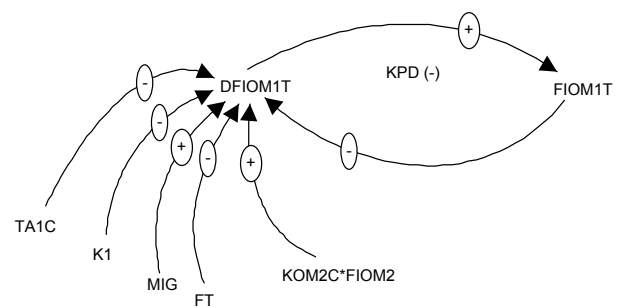


Figure1. Structural diagram of turbo compressor with low pressure

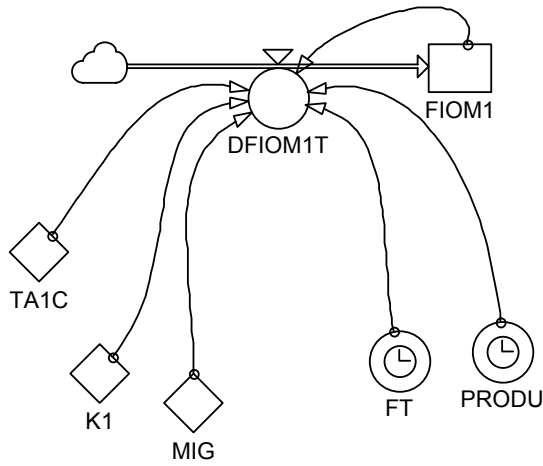


Figure 2. Flow diagram of turbo compressor with low pressure

In observed system there is one feed back loop (FBL):
FBL1(-): FIOM1=>(-) DFIOM1T=>(+)DFIOM1T=>
(+)FIOM1;
with self regulating dynamic character (-), because the
addition of negative sign is odd number.

Equations of shaft of turbo compressor with high pressure:

$$T_{a2}\varphi_{\omega 2} + k_2\varphi_{\omega 2} = \mu_G + k_{\omega 1}\varphi_{\omega 1} \quad (4)$$

$$\frac{d\varphi_{\omega 2}}{dt} = \frac{k_2}{T_{a2}} \left[\frac{\mu_G}{k_2} + \frac{k_{\omega 1}\varphi_{\omega 1}}{k_2} - \varphi_{\omega 2} \right] \quad (5)$$

$$\frac{d\varphi_{\omega 2}}{dt} = \frac{1}{T_{a2}} (\mu_G + k_{\omega 1}\varphi_{\omega 1} - \varphi_{\omega 2}) \quad (6)$$

2. SYSTEM DYNAMICS SIMULATING MODELLING OF SHIP SYNCHRONOUS GENERATOR

In this short paper, it is impossible to give a complete model; complete model (30 equations) has been presented in IASTED, Pittsburgh, 1998, 372-375, (7).

Where are:

Ψ_d =PSID - stator flux linkage in the d-axis,

r_s - stator resistance

x_s - stator reactance,

Ψ_q =PSIQ - stator flux linkage in the q-axis,

ω = OME - diesel-engine angular velocity (angular frequency),

Ψ_{ad} =PSAD - stator mutual flux linkage in the d-axis,

u_d - stator voltage in the d-axis,

Ψ_{aq} =PSAQ - stator mutual flux linkage in the q-axis,

u_q - stator voltage in the q-axis,

u - summary stator voltage,

Ψ_f = PSIF - rotor exciting flux linkage,

r_f - rotor exciting resistance,

u_f - rotor exciting voltage,

Ψ_{1d} = PS1D - damping coil flux linkage in the d-axis,

r_{1d} - damping coil resistance in the d-axis,

x_{1d} - damping coil reactance in the d-axis,

Ψ_{1q} = PS1Q - damping coil flux linkage in the q-axis,

r_{1q} - damping coil resistance in the q-axis,

x_{1q} - damping coil reactance in the q-axis,

r_L - load resistance,

x_L - load reactance,

M_e = MEL - generator electromagnetic moment,

i_d - stator current in the d-axis,

i_q - stator current in the q-axis,

i_f - rotor exciting current and

i - summary stator current.

System Dynamics Flow Diagram of Synchronous Generator Set with UNIREG-PID regulator shown in Figure 3.

3. COMPUTING SIMULATING MODEL OF GAS-TURBO GENERATOR

Simulation scenario

Run of the turbine is triple-stage, which mean that in TIME=1 second we accelerate turbine by bringing the fuel. On 10% of nominal number of revolution in TIME=5 seconds we increase fuel consumption to the 35% of nominal number of revolution, and in TIME=10 we increase fuel supply, and in that way obtain "uniform" heating of turbine i.e. nominal number of revolution. In TIME=20 PID regulator is involved, and in TIME=30 exciting of the generator start. In TIME=30 stochastically load occurs.

Listing of the program:

MACRO FUNCTIONS:

"FIRST DERIVATION-SPEED":

MACRO SLOPE(X,DEL

A SLOPE.K=(X.K-SMOOTH(X.K,DEL))/DT

MEND

" PID REGULATOR-UNIREG":

MACRO UNIREG(X,KPP,KPI,KPD

INTRN ID,PREG,IREG,DREG

A PREG.K=KPP*X.K

L ID.K=ID.J+DT*X.J

N ID=X

A IREG.K=KPI*ID.K

A DREG.K=KPD*SLOPE(X.K,DT)

A UNIREG.K=PREG.K+IREG.K+DREG.K

MEND

R DFIOM1T.KL=(1/TA1C)*(MIG.K+KOM2C*FIOM2.K-FT.K-K1*FIOM1.K)

C TA1C=.65

C KOM2C=.7

C K1=1.128

L FIOM1.K=FIOM1.J+DT*DFIOM1T.JK

```

N FIOM1=0
A MIG.K=STEP(.1/6,1)+STEP(.25/6,5)+
  STEP(.45/6,10)+STEP(PIDREG.K,20)
A DISK.K=CLIP(1-FIOM1.K,0,TIME.K,20)
A PIDREG.K=UNIREG(DISK.K,KPP,KPI,KPD)
C KPP=1
C KPI=1
C KPD=0
A FT.K=-MEL.K
R DFIOM2T.KL=(1/TA2C)*(MIG.K+KOM1C*
  FIOM1.K-K2*FIOM2.K)
C TA2C=.4
C KOM1C=1.616
C K2=1.2
L FIOM2.K=FIOM2.J+DT*DFIOM2T.JK
N FIOM2=0
R DPSIDDT.KL=((RS*PSID.K)/XS)+PSIQ.K*
  OME.K+((RS*PSAD.K)/XS)+UD.K
L PSID.K=PSID.J+DT*DPSIDDT.JK
N PSID=0
C RS=1
A UD.K=CLIP(RE.K*ID.K+XE*
  SLOPE(ID.K,DT)+XE*IQ.K*OME.K,0,TIME.K,30)
A RE.K=1+STEP(.2,60)*NOISE()
C XE=1
SAVE DPSIDDT,PSID,OME,UD
R DPSIQDT.KL=PSID.K*OME.K-((RS*PSIQ.K)/XS)
  +((RS*PSAQ.K)/XS)+UQ.K
L PSIQ.K=PSIQ.J+DT*DPSIQDT.JK
N PSIQ=0
A UQ.K=CLIP(RE.K*IQ.K+XE*SLOPE(IQ.K,DT)+
  XE*ID.K*OME.K,0,TIME.K,30)
A U.K=SQRT(UD.K*UD.K+UQ.K*UQ.K)
SAVE DPSIQDT,PSIQ,UQ
R DPSIFDT.KL=((RF*PSIF.K)/XF)+
  ((RF*PSAD.K)/XF)+UF.K
L PSIF.K=PSIF.J+DT*DPSIFDT.JK
N PSIF=0
C RF=1
SAVE DPSIFDT,PSIF
R DPS1DDT.KL=((RID*PSID.K)/X1D)+
  ((RID*PSAD.K)/X1D)
L PS1D.K=PS1D.J+DT*DPS1DDT.JK
N PS1D=0
C RID=1
SAVE DPSIDDT,PSID
R DPS1QDT.KL=((R1Q*PS1Q.K)/X1Q)+
  ((R1Q*PSAQ.K)/X1Q)
L PS1Q.K=PS1Q.J+DT*DPS1QDT.JK
N PS1Q=0
C R1Q=1
SAVE DPSIQDT,PSIQ
A PSAD.K=X1*((PSID.K/XS)+(PSIF.K/XF)+
  PSID.K/X1D)
C XAD=1
C XS=1
SAVE PSAD,PSAQ
A PSAQ.K=X2*((PSIQ.K/XS)+(PSIQ.K/X1Q))

```

```

C XAQ=1
C XF=1
C X1D=1
C X1Q=1
A OME.K=FIOM1.K
N OME=0
A MEL.K=(PSID.K*PSAQ.K-PSIQ.K*PSAD.K)/XS
SAVE MEL,OME
A UF.K=UNIREG(DISK1.K,KPP1,KPI1,KPD1)
A DISK1.K=CLIP(1-U.K,0,TIME.K,30)
C KPP1=.6
C KPI1=0
C KPD1=.1
K X1=1/((1/XAD)+(1/XS)+(1/XF)+(1/X1D))
K X2=1/((1/XAQ)+(1/XS)+(1/X1D))
SAVE U,UF
A ID.K=CLIP(-(PSID.K-PSAD.K)/XS,0,TIME.K,30)
A IQ.K=CLIP(-(PSIQ.K-PSAQ.K)/XS,0,TIME.K,30)
A I.K=SQRT(ID.K*ID.K+IQ.K*IQ.K)
A IF.K=CLIP(-(PSIF.K-PSAD.K)/XS,0,TIME.K,30)
SAVE ID,IQ,IF,I
SAVE FIOM1,DFIOM1T,FIOM2,DFIOM2T,MIG,FT
SPEC DT=.1,LENGTH=100,SAVPER=1

```

Graphics results of the simulation:

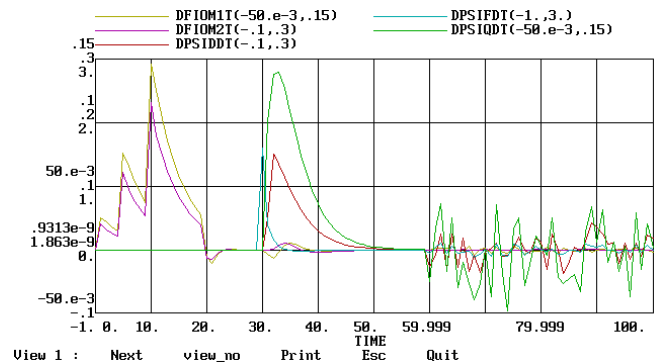


Figure 4. Speed variation of relative angular speed, speed variation of relative angular speed of second shaft, speed variation of stator flux linkage in the d-axis, q-axis and speed variation of rotor exciting flux linkage

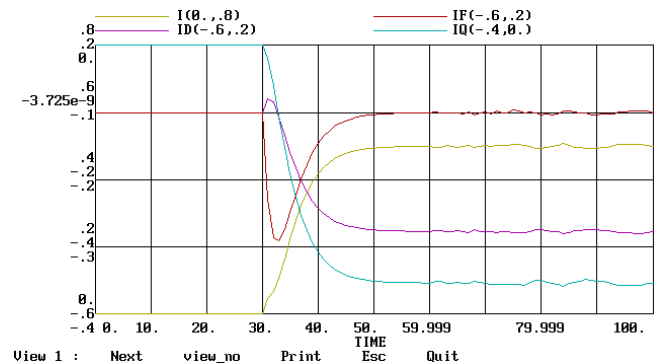


Figure 5. Stator current in the d-axis, stator current in the q-axis, rotor exciting current and summary stator current

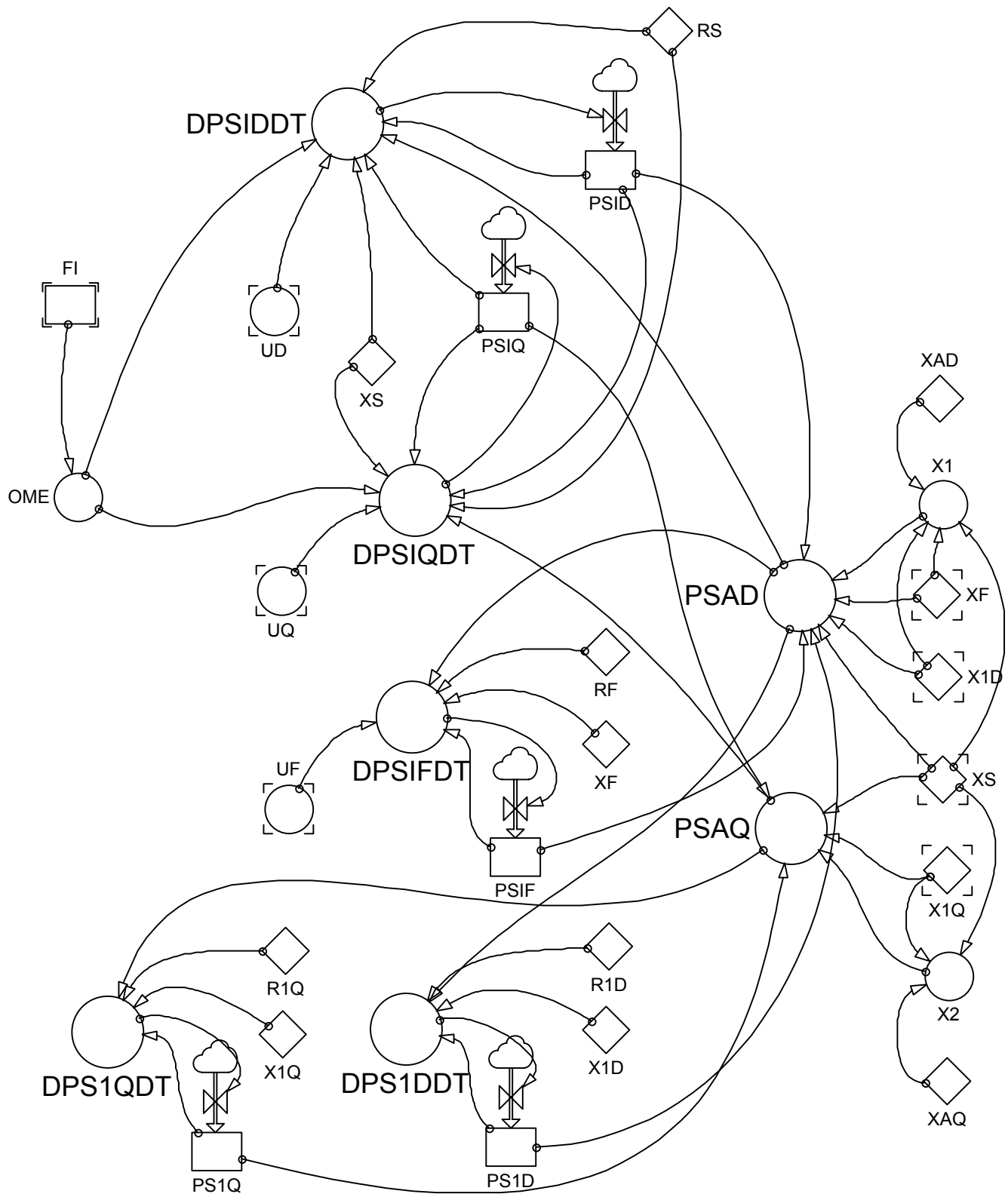


Figure 3. System Dynamics Flow Diagram of Synchronous Generator Set with UNIREG-PID

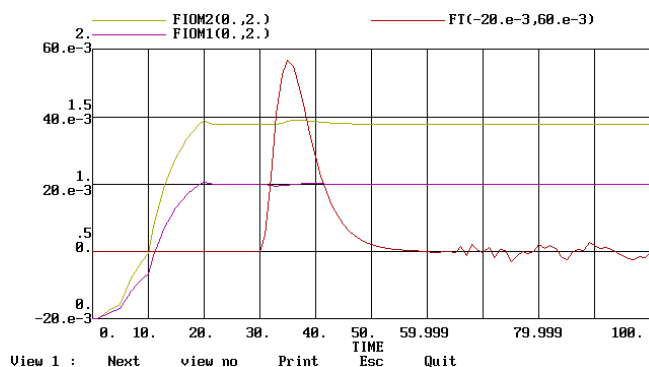


Figure 6. Relative angular speed variation, relative angular speed variation of second shaft,

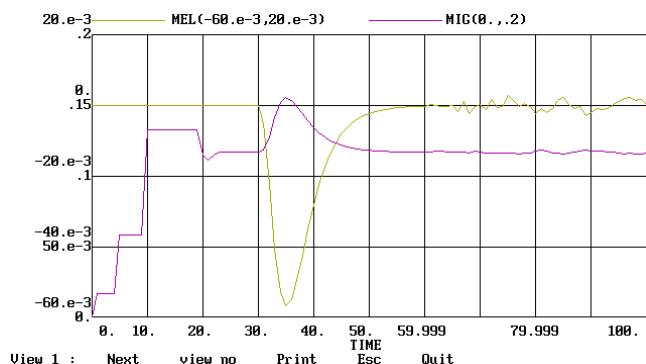


Figure 7. Generator electromagnetic moment, relative variation of fuel consumption

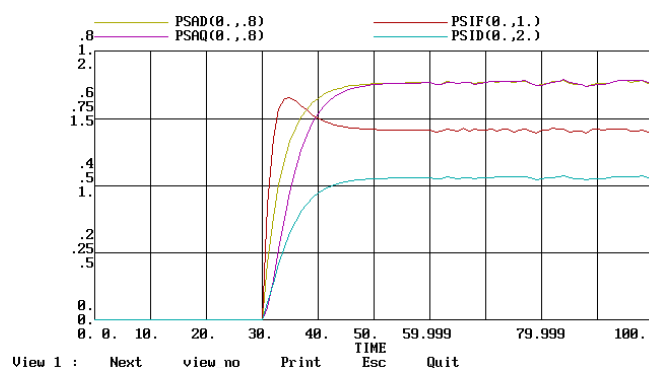


Figure 8. Stator flux linkage in the d-axis, stator mutual flux linkage in the d-axis, rotor exciting flux linkage, stator mutual flux linkage in the q-axis

After turn of the turbine in idling PID regulator is involved. PID regulator acts on fuel supply and with this act brings number of revolution of turbine on nominal value. Switch on the synchronous generator increase fuel consumption, while the number of revolution of both shafts varies relatively small. In regime of stochastically load (we simulate alternating switch on and off of electrical consumer generating random number) attempt to stabilizes output voltage with self-excited according to the phase compounding scheme was not effective enough. After replacing this part of assembly with electrical PID regulator

it is visible from diagram, those output voltages vary in permissible limits, although variations of load are extremely unfavorable.

4. CONCLUSION

The application of System Dynamics Simulation Modelling Approach of the complex marine dynamic processes, which the authors, together with their graduate students, carried out at the Maritime Faculty University of Split - Croatia seven years ago, revealed the following facts:

1. The System Dynamics Modelling Approach is a very suitable software education tool for marine students and engineers.
2. System Dynamics Computer Simulation Models of marine systems or processes are very effective and successfully implemented in simulation and training courses as part of the marine education process.

Based on our long term experience in the application of the dynamical methodology of simulating and in this short presentation we provide every expert in need with the possibility to acquire additional knowledge about the same system in a quick scientifically based way of exploring the complex systems. It means:

Do not simulating by the "black box" way, it would be better if they would do simulating by this "white box" approach!

REFERENCES

- Forrester, Jay W. 1973/1971. *Principles of Systems*, MIT Press, Cambridge Massachusetts, USA, 1973.
- Munitic, A., Milic M. and Milikovic M. 1997. System Dynamics Computer Simulation Model of the Marine Diesel-Drive Generation Set, *World Congress on Scientific Computation, Modelling and Applied Mathematics*, Berlin, 1997.
- Munitic, A. 1989. *Computer Simulation with Help of System Dynamics*, BIS, Croatia, 1989.
- Miler, J., 1955, *Parne i plinske turbine*, Zagreb, Croatia, 1955.
- Richardson, George P. and Pugh III Aleksander L. 1981. *Introduction to System Dynamics Modelling with Dynamo*, MIT Press, Cambridge, Massachusetts, USA, 1981.
- He: eB4ha, P. A., 1975, Abtomat4La*4β cP*ob.: ix _hepnet4Peck4x Pctahobok, 1975.,
- Munitic, A., Kuzmanic, I., Krcum, M., 1998., System Dynamic Simulation Modelling of the Marine Synchronous Generator, *Modelling and Simulation, IASTED*, Pittsburgh, 1998, 372-375.
- Munitic, A., et al.. 2002. System Dynamics Modelling of Complex Electro Mechanical System, *IASTED, AMS 2002*, Cambridge, USA, 2002, 511-515.
- Munitic, A., Kulenovic, Z., Dvornik, J., 2003., Computing Simulation of Driving System- "Ship/Piston Compressor-Electric motor", *IASTED MS 2003*, California, USA, 2003, 515-520.

AUTOMOTIVE SIMULATION

Parts Simulation

CONTINUOUS SIMULATION OF ACCELERATION PERFORMANCE OF A CAR EQUIPPED WITH AN AUTOMATIC GEARBOX

Olivier Verlinden
Georges Kouroussis
Calogero Conti
Faculté Polytechnique de Mons
Boulevard Dolez 31
7000 Mons (BELGIUM)
Olivier.Verlinden@fpms.ac.be

Joel Crape
David Vercouter
AWTC Europe
Avenue de l'Industrie, 19
1420 Braine l'Alleud (BELGIUM)

KEYWORDS

Automotive vehicle, automatic gearbox, acceleration, dynamics, differential equations

ABSTRACT

This paper presents the model of an automatic gearbox allowing a continuous simulation of the acceleration process of an automotive vehicle, including the gearshifts. In an automatic gearbox, the gear ratio is chosen by engaging/disengaging internal clutches and brakes which change the kinematic constraints between the rotating parts. The proposed model involves all rotating parts of the gearbox but introduces the torques transmitted by the clutches/brakes through a stabilized penalty function related to the kinematic constraint associated with the device. A set of penalty coefficients is associated to each gear ratio. During a gear shift, the penalty coefficients progressively switch from one set to the next one, according to the law actually imposed by the gearbox controller. Such a model has the advantage of the simplicity as it doesn't modify the nature of the equations of motion. The model has been used for the simulation of the acceleration of a passenger car, including the motor and torque converter characteristics, a realistic tire model and classical road loads like rolling resistance and the aerodynamic drag. The results have been positively compared to a measurement campaign performed by the constructor.

INTRODUCTION

Modern passenger cars are more and more often equipped with automatic gearboxes. Such a device improves the comfort of the driver, especially on long distances, but is also safer for the power train in case of powerful engines. Moreover, the gear shifts can be optimized by the constructor for different purposes. For a low throttle position, the gear shifts will be optimized for a minimal fuel consumption while the acceleration performance will be aimed at for a full throttle position. To help in the design of the gearshifts, it is interesting to dispose of a model allowing the simulation of the complete acceleration process of the vehicle. In a near future, the model could even be integrated inside the gearbox controller.

Although multibody system simulation tools like ADAMS,

SIMPACT or LMS/DADS are now commonly used to assess the dynamic performances of a vehicle, the simulation of a complete acceleration process is not straightforward as each gear shift changes the kinematic constraints imposed by the drivetrain. To simulate the gear shift, a first option [Pan and Moskwa 1995] consists in building a model for the successive gear ratios, the simulation jumping from model to model according to some continuity conditions. A second option is to model the behaviour of the internal components of the automatic gearbox, where the gear ratio is practically controlled by locking/unlocking clutches and/or brakes driven by hydraulic actuators. In [Otter et al. 1997], the switch between different sets of kinematic constraints is automatically guided by the friction limit: a clutch or a brake is engaged or slipping depending on whether the transmitted torque is below the friction limit. Eventually, a still more sophisticated model could be developed, involving all mechanical parts of the gearbox, and detailing the contact phenomena occurring in clutches and brakes [Rahnejat 1998]. Such a model is especially useful for the mechanical design of the gearbox.

We propose here an intermediary model, involving all rotating parts of the gearbox, but modelling the torques transmitted in a clutch/brake through penalty functions related to the kinematic constraint associated with the device. After a brief presentation of its working, the kinematic model of the gearbox will be developed. The model chosen to incorporate the clutches and brakes will then be detailed, and the motion equations of the gearbox will be derived. The latter will be used to set up the equations governing the longitudinal behaviour of a passenger car. The model will finally be applied to an actual vehicle and the results compared to measurements performed by the constructor.

WORKING OF THE GEARBOX

The layout of the considered gearbox (AISIN 55-50 SN) is illustrated in figure 1. It is composed of eighteen parts (S_1 to S_{18}), S_1 and S_{18} being respectively the input and driven shafts. Parts S_2 to S_8 constitute the main planetary, also classically called Ravigneaux planetary. Parts S_{11} to S_{18} among which the driven shaft (S_{18}) constitute the underdrive planetary.

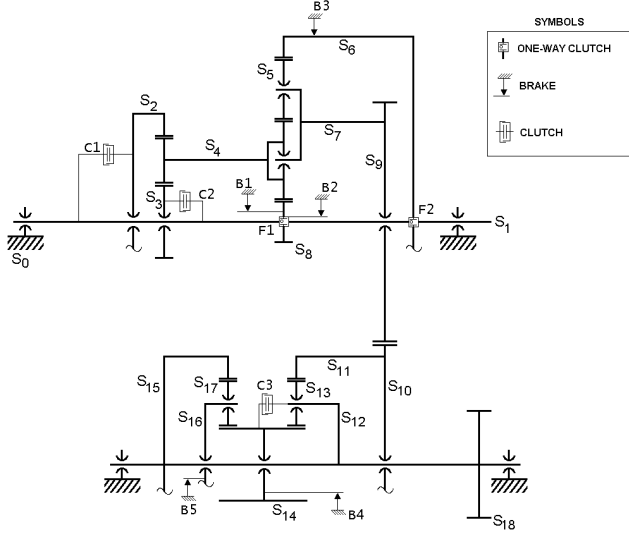


Figure 1: Layout of the gearbox

Practically, the power flows from the main planetary to the underdrive planetary through the joint between the counter drive gear S_9 and the counter driven gear S_{10} . The gear ratio between the input shaft S_1 and the counter drive gear S_9 will depend on the state of clutches $C1$ and $C2$ and brakes $B1$, $B2$ and $B3$ ($F1$ and $F2$ are free wheels). On the other hand, the gear ratio between the counter driven gear S_{10} and the driven shaft S_{18} depends on the state of clutch $C3$ and brakes $B4$ and $B5$. All brakes and clutches are equipped with hydraulic actuators driven by a central unit which controls the wanted global gear ratio.

The major idea of the model presented hereafter is to consider the gearbox with unlocked devices and to add the effect of clutches and brakes through external torques.

KINEMATIC MODEL OF THE GEARBOX

The first step to perform the kinematic analysis of the gearbox consists in determining the number of degrees of freedom. In our case, only the rotation of each part about its central axis has to be considered. The number of rotating parts is equal to 18, subjected to 14 constraints coming from 4 assemblies (S_7 - S_9 , S_{10} - S_{11} , S_{12} - S_{18} and S_{15} - S_{18}) and 10 gearings (S_2 - S_4 , S_3 - S_4 , S_4 - S_5 , S_4 - S_8 , S_5 - S_6 , S_9 - S_{10} , S_{11} - S_{13} , S_{13} - S_{14} , S_{14} - S_{17} and S_{15} - S_{17}) which leads to a number of 4 degrees of freedom. As the clutch $C1$ is always engaged in normal drive mode, there remain 3 degrees of freedom.

We have chosen to express the rotation velocity of each part from the ones of parts S_1 , S_9 and S_{18} . The work can be achieved by writing for each gearing the well-known relationship (cf. figure 2)

$$\frac{\omega_{i/k}}{\omega_{j/k}} = -\frac{Z_j}{Z_i} \quad (1)$$

with $\omega_{i/k}$ the rotation velocity of gear S_i with respect to S_k

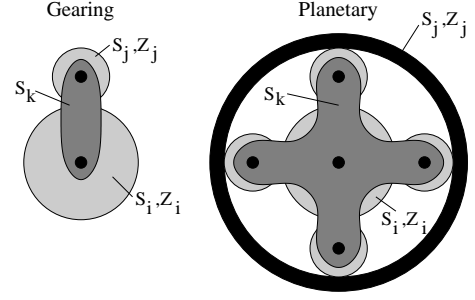


Figure 2: Gearing and planetary

(which is the part the gear axes are attached to) and Z_i the number of teeth of S_i . This relationship must be combined with the composition of rotations

$$\omega_{i/k} = \omega_{i/j} + \omega_{j/k} \quad (2)$$

For a planetary, the Willis formula [Hulin 1990] can be used directly (cf. figure 2)

$$\frac{\omega_{j/0} - \omega_{k/0}}{\omega_{i/0} - \omega_{k/0}} = -\frac{Z_i}{Z_j} \quad (3)$$

where the reference part 0 can be replaced by any one, according to the composition of rotations.

After all calculations, we get for example for part S_{13}

$$\omega_{13} = \left(-\frac{Z_9}{Z_{10}} \frac{Z_{11}}{Z_{13}}\right)\omega_9 + \left(1 - \frac{Z_{11}}{Z_{13}}\right)\omega_{18} \quad (4)$$

MODELLING OF CLUTCHES AND BRAKES

To assure the power transmission from input to output shaft, two supplementary devices, either a clutch or a brake, must be engaged so that the gearbox owns only one degree of freedom. If a clutch between parts S_i and S_j is engaged, it will progressively prevent the relative rotation speed, imposing the following constraint

$$\omega_i - \omega_j = 0 \quad (5)$$

Practically, a friction torque arises through the friction disk between the input and output shafts of the clutch, the maximum amplitude depending on the pressure on the friction disk. The pressure progressively increases and, once the transmitted torque falls below the friction limit, the disks stick to each other and the constraint holds. The smoothness of the gear change is directly dependent on the pressure profile imposed by the hydraulic actuator, driven itself by the gearbox controller.

To allow a continuous and simple simulation of the gear change, a clutch will be introduced into the equations of motion by applying torques T_i and T_j respectively on parts S_i and S_j , given by

$$T_i = -T_j = K (\omega_j - \omega_i + \tau(\dot{\omega}_j - \dot{\omega}_i)) \quad (6)$$

During a gear shift, the penalty coefficient K follows the pressure profile imposed by the control unit and the simulation can proceed continuously. The time constant τ is introduced to insure the stability of the numerical integration so that the model can also be seen as some kind of PD control of the related kinematic constraint. Of course, some slip occurs in the device but is negligible if the penalty coefficient is sufficiently large, which is possible thanks to the stabilization term. The case of a brake is similar, one of the parts being fixed.

MOTION EQUATIONS OF THE GEARBOX

From the kinematics and the clutch/brake torques, the 3 equations of motion of the gearbox can be easily obtained by means of the principle of virtual power as

$$\sum_{S_i} (I_i \dot{\omega}_i - T_{tot,i}) \frac{\partial \omega_i}{\partial \omega_1} = 0 \quad (7)$$

$$\sum_{S_i} (I_i \dot{\omega}_i - T_{tot,i}) \frac{\partial \omega_i}{\partial \omega_9} = 0 \quad (8)$$

$$\sum_{S_i} (I_i \dot{\omega}_i - T_{tot,i}) \frac{\partial \omega_i}{\partial \omega_{18}} = 0 \quad (9)$$

with I_i the axial moment of inertia of part S_i and $T_{tot,i}$ the total torque exerted on part S_i by brakes, clutches or external sources (for input and output shafts).

By developing the expression of $\dot{\omega}_i$ in terms of the master rotation accelerations ω_1 , ω_9 and ω_{18} , the following system is obtained

$$[\mathbf{M}] \begin{pmatrix} \dot{\omega}_1 \\ \dot{\omega}_9 \\ \dot{\omega}_{18} \end{pmatrix} + [\mathbf{C}] \begin{pmatrix} \omega_1 \\ \omega_9 \\ \omega_{18} \end{pmatrix} = \begin{pmatrix} T_{inp} \\ 0 \\ T_{out} \end{pmatrix} \quad (10)$$

with T_{inp} and T_{out} the torques applied on the input and output shafts. It is of interest to note that matrices $[\mathbf{M}]$ and $[\mathbf{C}]$ evolve with the value of the penalty coefficients relative to each clutch/brake.

These equations have the advantage to remain the same during the whole simulation. To each gear ratio corresponds a set of penalty coefficients and, during a gear shift, the penalty coefficients progressively switch from one set to the following. To be as close as possible to reality, the penalty coefficients follow, during the switch, the evolution of the pressure imposed by the gearbox controller in the hydraulic actuator driving the clutch/brake.

SIMULATION OF THE COMPLETE VEHICLE

The proposed model of the gearbox has been used to simulate the acceleration process of an actual vehicle. The power flow in the transmission (figure 3) shows that the simulation of the complete vehicle involves two more degrees of freedom

- the own rotation of the engine driving the rotation of the input shaft of the torque converter, also called the pump

shaft (the output shaft of the converter or turbine shaft is attached to the input shaft of the gearbox);

- the longitudinal motion of the vehicle itself, kinematically independent from the wheel rotation due to the tire slip (cf. later).

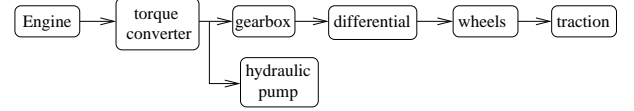


Figure 3: Power flow in the transmission

The dynamic equilibrium of the motor shaft is written

$$I_0 \dot{\omega}_0 = T_0 - T_{TC,pump} \quad (11)$$

with ω_0 the rotation velocity of the engine, I_0 the total moment of inertia related to the motor shaft including the motor itself but also the pump shaft of the torque converter, T_0 the motor torque and $T_{TC,pump}$ the torque transmitted to the pump shaft of the torque converter.

For the full throttle position which will be considered in the example, the motor torque T_0 can be estimated from a second-order polynomial, as

$$T_0 = m_2 \omega_0^2 + m_1 \omega_0 + m_0 \quad (12)$$

The torque $T_{TC,pump}$ driving the input shaft of the torque converter depends on the speed ratio between the two shafts of the converter. In our case, the rotation speed of the turbine shaft corresponds to ω_1 (rotation speed of the input shaft of the gearbox), and the torque can be approximated by

$$T_{TC,pump} = f_2 \omega_1^2 + f_1 \omega_1 \omega_0 + f_0 \omega_0^2 \quad (13)$$

In the complete vehicle, the torque exerted on the input shaft of the gearbox will be given by

$$T_{inp} = T_{TC,turb} - T_{hydr} - I_{1,ext} \dot{\omega}_1 \quad (14)$$

with $T_{TC,turb}$ the torque exerted by the turbine shaft of the torque converter, T_{hydr} the torque necessary to drive the hydraulic pump of the gearbox and $I_{1,ext}$ the moment of inertia related to all external elements connected to the input shaft of the gearbox like the torque converter or the hydraulic pump.

The torques related to the converter and the hydraulic pump can be approximated by

$$T_{TC,turb} = T_{TC,pump} \left(t_1 \frac{\omega_1}{\omega_0} + t_0 \right) \quad (15)$$

$$T_{hydr} = a \omega_1 + b \quad (16)$$

t_1 and t_0 depending on the converter type, and a, b depending on the pump and the circuit pressure.

The equation of equilibrium related to the longitudinal motion of the car is written

$$M\ddot{x} = 2F_t - R_r - D_A \quad (17)$$

with M the mass of the vehicle, F_t the traction force of a tire, R_r the rolling resistance and D_A the aerodynamic drag.

The traction force depends on the tire slip s defined by

$$s = \frac{\omega_w R_w - \dot{x}}{\dot{x}} = \frac{r_d \omega_{18} R_w - \dot{x}}{\dot{x}} \quad (18)$$

with R_w the rolling radius of the wheel and r_d the transmission ratio between the output shaft of the gearbox and the wheel.

For each front wheel, the traction force is evaluated from the model of the University of Arizona [Gim 1988].

$$F_t = C_L s - \frac{C_L^2}{3fF_z} s^2 + \frac{C_L^3}{27f^2 F_z^2} s^3 \quad \text{if } s < s_{lim} \quad (19)$$

$$= fF_z \quad \text{if } s \geq s_{lim} \quad (20)$$

with C_L the so-called longitudinal stiffness of the tire (equivalent to the cornering stiffness in lateral), f the ground-tire friction coefficient, F_z the vertical force on the tire, and s_{lim} the limit slip, defined later.

It is of interest to note that the vertical force F_z on the front wheels decreases with the acceleration

$$F_z = \frac{M}{2} \left(\frac{gc}{L} - \frac{\ddot{x}h}{L} \right) \quad (21)$$

with L the wheelbase of the vehicle, c the longitudinal distance between the center of gravity and the rear axle and h the height of the center of gravity with respect to the ground.

The friction coefficient f evolves from the static friction coefficient f_s to the dynamic friction coefficient f_d according to the slip

$$f = f_s - (f_s - f_d) \frac{s}{s_{lim}} \quad \text{with } s_{lim} = \frac{3f_d F_z}{C_L} \quad (22)$$

In a general way, the rolling resistance R_r is estimated through the rolling resistance coefficient f_r , by

$$R_r = f_r M g \quad (23)$$

The coefficient f_r depends on the tire type, on the road roughness and on velocity V . For a conventional road, the following formula yields a good estimation [Gillespie 1992]

$$\begin{aligned} f_r &= 0.0001(120 + \frac{V(\text{km/h})}{8}) \quad \text{if } V \leq 110 \text{ km/h} \\ &= 0.0001(140 + 0.625(V(\text{km/h}) - 120)) \\ &\quad \text{if } V > 110 \text{ km/h} \end{aligned} \quad (24)$$

The aerodynamic drag D_A is classically estimated from the drag coefficient C_D (also called C_X) of the vehicle by

$$D_A = \frac{1}{2} \rho C_D A V^2 \quad (25)$$

with ρ the air density, A the frontal area of the vehicle and V the speed. The coefficient C_D is classically a bit lower than 0.3 for a passenger car.

The torque to exert on the output shaft of the gearbox is given by

$$T_{out} = -I_{18,ext} \dot{\omega}_{18} - 2r_d R_w F_t \quad (26)$$

with $I_{18,ext}$ the equivalent moment of inertia for all parts of the transmission from the gearbox to the wheels. The major contribution comes from the wheels, given by $2I_w r_d^2$ with I_w the moment of inertia of the wheel with respect to its rotation axis.

The 5 equations governing the motion of the complete vehicle then have the following form

$$[M] \begin{pmatrix} \dot{\omega}_0 \\ \dot{\omega}_1 \\ \dot{\omega}_9 \\ \dot{\omega}_{18} \\ \dot{x} \end{pmatrix} + [C] \begin{pmatrix} \omega_0 \\ \omega_1 \\ \omega_9 \\ \omega_{18} \\ x \end{pmatrix} = \begin{pmatrix} T_0 - T_{TC,pump} \\ T_{TC,turb} - T_{hydr} \\ 0 \\ -2r_d R_w F_t \\ 2F_t - R_r - D_A \end{pmatrix} \quad (27)$$

Due to the chosen model, the effect of clutches and brakes are automatically included in matrices $[M]$ and $[C]$. It is of interest to note that the motion equations are highly stiff, due to the high value of the penalty coefficients.

SIMULATION RESULTS

The acceleration process of a given passenger car has been simulated. The data have been provided by the constructor and cannot be given here for confidentiality reasons. The constructor also performed some measurements that will be used for validation of the model. The gear shifts were also defined by the constructor. They depend primarily on the velocity and the position of the acceleration pedal but also on different measurements performed on the vehicle.

Concerning the treatment of clutches and brakes, a penalty coefficient equal to 10^5 has been used, the constant time τ being equal to 0.1 s.

The equations of motion are integrated with the help of the EasyDyn library [Verlinden et. al 2003], dedicated more especially to second-order, and eventually stiff, differential equations. The numerical integration is based on the Newmark formulas. The library is written in C++ and is open source as well as the GSL (GNU Scientific Library) it relies on, to avoid the dependence on any commercial editor, and to allow the portability and the usability in the controller itself.

Some results of the simulation are shown in figures 4 to 7 and compared to the measurements performed by the constructor. During the experimental test, the vehicle reached the end of the track after 20 s and the test had to be stopped. Globally, the diagrams show a good agreement between measured and

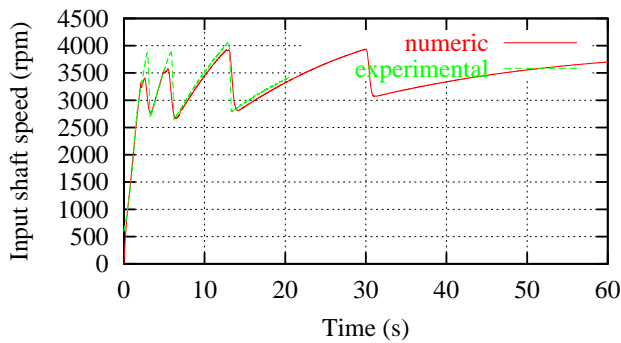


Figure 4: Input speed of the gearbox

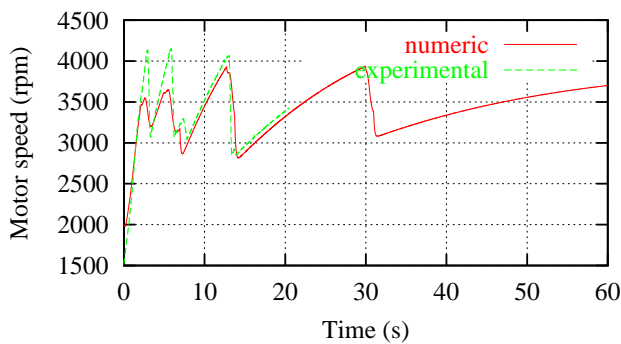


Figure 5: Motor speed

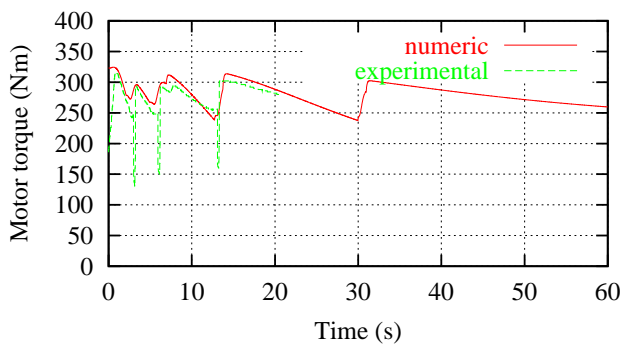


Figure 6: Motor torque

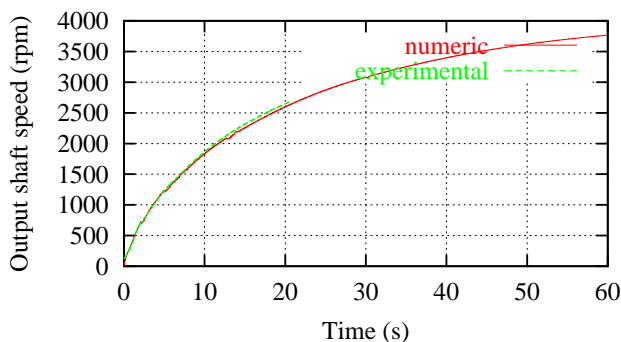


Figure 7: Output speed of the gearbox

simulated results. The major difference occurs during the gear shift itself, for two principal reasons

- the model used for the torque converter considers only stationary conditions, which is obviously not the case during the gear shift and even globally speaking for the first two gear ratios;
- the simulation does not take into account the release of the engine throttle during the gear shift (cf. figure 6).

CONCLUSION

A model has been presented for the continuous simulation of the gear shifts of an automatic gearbox. The idea is to introduce the kinematic constraints imposed by the internal clutches/brakes through a stabilized penalty formulation. Such a model is not intended for a study of the gearbox itself but allows a fluent simulation of a vehicle acceleration in order to tune the gear shifts to the considered vehicle.

The main advantage of the model is its simplicity, although it can be a good starting point for a more sophisticated study involving a finer representation of the transient behaviour of the clutches or brakes.

The model has been used to simulate the acceleration of an actual passenger car, showing a good agreement with the measurements performed by the constructor.

References

- [Gim 1988] Gim G., 1988, *Vehicle Dynamic Simulation with a Comprehensive Model for Pneumatic Tires*, PhD, University of Arizona.
- [Hulin 1990] Hulin R., 1990, "Boîtes de vitesses", Ed. Techniques de l'ingénieur, traité Génie mécanique B5 660.
- [Otter et al. 1997] Otter M., Schlegel C. & Elmqvist H., 1997, "Modeling and Realtime Simulation of an Automatic Gearbox using Modelica", *Proceedings of ESS'97*, Oct. 19-22, Passau (Germany), pp 115-121.
- [Pan and Moskwa 1995] Pan C.-H. & Moskwa J. J., 1995, "Dynamic modeling and Simulation of the Ford AOD Automobile Transmission", SAE paper 950899.
- [Rahnejat 1998] Rahnejat H., 1998, *Multi-body Dynamics. Vehicles, Machines and Mechanisms*, SAE, Warrendale, PA.
- [Gillespie 1992] Gillespie T. D., 1992, *Fundamentals of Vehicle Dynamics*, SAE, Warrendale, PA.
- [Verlinden et. al 2003] Verlinden O., Kouroussis G., Datoussaid S. & Conti C., 2003, "Open Source Symbolic and Numerical Tools for the Simulation of Multi-body Systems", *Proceedings of the National Congress on Theoretical and Applied Mechanics*, May 26-27, Ghent.

SIMULATION VERIFICATION OF FAULT-TOLERANT BRAKE-BY-WIRE SYSTEM

Petr Grillinger
University of West Bohemia, Department of Computer Science
Univerzitni 22, Box 314
30614 Plzen, Czech Republic
E-mail: pgrillin@kiv.zcu.cz

KEYWORDS

C-Sim, TTA, TTP/C, Fault tolerance, Fault injection.

ABSTRACT

This paper presents executable simulation model of a Volvo car equipped by brake-by-wire technology based on a state-of-art communication protocol — the time triggered TTP/C protocol. The model is written in C language and uses C-Sim discrete-time simulation library. The aim of the model and of this paper is to verify fault tolerance of the braking process and to examine possible impacts of a failure. Two methods are applied to give a convincing proof of the safety of braking mechanism — simulated fault injection and online visualization. Statistical data measured during fault injection experiments are included in results as well as a sample screen shot from the visualization. Both methods yield results that show very high degree of safety achieved at architectural, communication and application level.

INTRODUCTION

In the future, increasing number of systems will move from mechanical and hydraulic control systems to electronic microcontrollers interconnected by a communication network (so called smart sensors/actuators). Until now this technology is deployed in larger scale mostly in non-safety critical applications (e.g. using CAN). Safety critical applications (as x-by-wire in cars) require ultra-reliable systems that are expensive. It is possible and probably safe to use a solution from airplane industry (e.g. *ByteFlight*) in a car, but not necessarily optimal and cost-effective. A solution for automotive industry is emerging in the form of a new generation of safety-targeted protocols (e.g. *FlexRay* and *TTP/C*). This paper presents simulation model of a Volvo car equipped with TTP/C communication system (Kopetz 1997). A basic overview of TTP/C protocol and the car model is given below.

TTP/C Communication Protocol

The TTP/C protocol (Time Triggered Protocol for SAE class C requirements) was designed by Prof. Kopetz to meet the specific needs for a cost-effective fault-tolerant real-time communication system. Currently two chip implementations for TTP/C microcontrollers are available — C1 and C2 (both are produced by TTTech, Austria). The C1 implementation has been extensively tested during past

years with an EU 5th framework program project (FIT). The results of the project proved in general the safety of TTP/C protocol. Research group from Czech Republic performed fault injection experiments using an executable simulation model of the communication protocol. The braking model presented in this paper is based on the model and tools developed within the project (web pages at <http://www.fit.zcu.cz>).

The communication protocol is based on *time division multiple access* method (TDMA) instead of the more common CSMA method (used by CAN). Every node that participates in communication has a statically allocated sending slot within every communication round (the basic communication period is called TDMA round) and may not send outside this slot. The protocol uses two independent channels for broadcast transmissions, provides synchronized global time and supports active node replication (this restricts the behavior of replicated nodes to be strictly deterministic). Failed nodes can be repaired and reintegrated without any impact on the operation of other nodes. Every node consists of two parts — the communication controller handles the transmission of messages and provides the basic TTP/C services; the host processor executes a user defined task-based application. These two parts are connected by a dual-port memory interface (called *CNI*).

C-Sim Based Simulation Model

C-Sim is a low-level simulation library written for maximum portability entirely in ANSI C. It provides means for pseudo-parallel process execution in discrete time (the mechanism is based on co-routines). The most important features of C-Sim are the deterministic scheduling of processes (this implies a total repeatability of any experiments) and its performance (web pages <http://www.c-sim.zcu.cz>).

The TTP/C protocol model is split into several layers:

- *Functional layer* describes pure functionality of the protocol, i.e. provides data structures and operations upon them.
- *Simulation layer* glues appropriate functions from the functional layer into executable C-Sim processes.
- *Application layer* connects different simulation processes to form a working application. The configuration of every application (number of nodes, programming of the hosts) depends on its purpose.

Car Model

The simulation model requires a working model of a car. We used a computation model of ABS braking system and a mechanical model of car movement, both programmed by Volvo Technology Corp. (Lennevi 2001, Lönn 2001). The movement model was designed to run in discrete time (simulation progresses by a specified amount of time units every time a single dedicated function is called). The interface of the model is exactly specified by a set of input and output variables and a model-progressing function, which must be called with a fixed period of model time (here equivalent to 0.0005 sec).

Input variables:

- Brake force to apply at every wheel (4 times)
- Traction torque for every wheel (4 times)
- Surface adhesion for every wheel (4 times)
- Steering wheel angle

Output variables:

- Every single wheel speed (4 times)
- Vehicle velocity
- Vehicle heading angle
- Position in X axis
- Position in Y axis

The ABS braking model can be expressed mathematically as a simple function $f(W, B, F_w, S_w)$ that transforms brake pedal force B to brake force that should be applied at wheel W . The function requires furthermore the current speed S_w of wheel W and fault vector image F_w (meaning of fault vector will be explained later) of node W .

BRAKE-BY-WIRE APPLICATION

The two described car models and TTP/C model were joined to form a single application called *BBW4* (Brake-by-wire 4 wheels; there is also a version that simulates braking of a single wheel). A conceptual model of the application (basic units and interactions) is presented in figure 1. There are three units that accept external input signals — the brake pedal, acceleration pedal and steering wheel. The later two are connected directly to the vehicle model and do not use the TTP/C subsystem. Brake pedal is connected to two TTP/C nodes (replica determinae) that measure force applied to the pedal. A basic loop of TTP/C based braking system looks as follows:

- Wheel nodes send currently applied braking force.
- Wheel nodes compute a *fault vector* that contains 1 for every wheel that operates correctly (at least one correct message was received).
- Brake pedal nodes measure brake pedal force and send the requested value of brake force.
- Wheel nodes compute local braking force using the ABS function f , requested brake force, fault vector and locally measured wheel speed (wheel speed is read from vehicle model, local brake force is applied to the model).

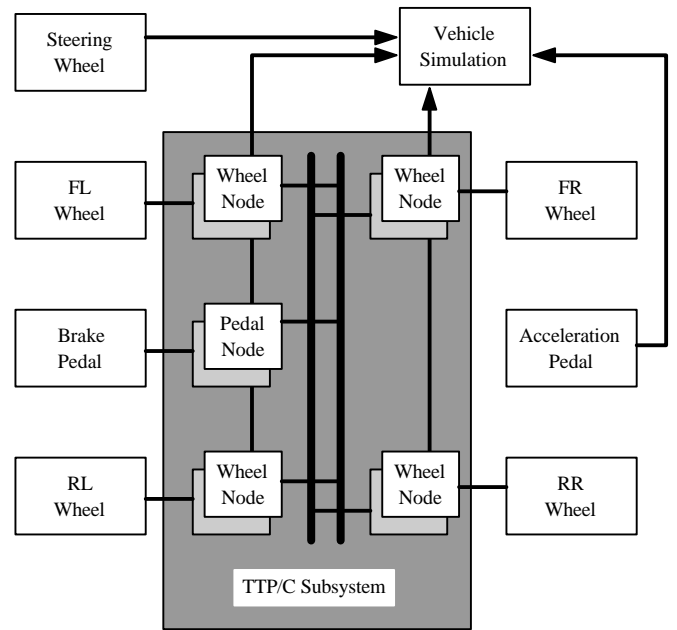


Figure 1: Model of TTP/C Based Braking Simulation

All TTP/C nodes used in this setup are replicated. Because TTP/C protocol offers the *fail-silence* feature (nodes send correct results or none at all), node replication ensures that the whole unit will be operational even if one of the nodes fails. The programming of all wheel nodes is the same. The duration of single TDMA round (basic communication period) was set to 2 ms, which is sufficient to transmit all messages using a 1 Mbit/sec bus.

Traction torque for every wheel is computed using a simple model (non-realistic) since it is not essential for the braking process (we assume that the car engine is passive during braking). It is used only to accelerate the car until a set velocity is reached and then the braking starts. The problem of steering the vehicle while the car brakes is too complex and also unnecessary, so no steering is allowed — the car's front wheels are always in their neutral position.

Visual interface

The output of the vehicle model is sufficient to create a complete visualization of the car's movement — screen shot of such visualization is presented in figure 2. This particular interface was written in C++ Builder and allows anyone to easily monitor the vehicle behavior and to perform basic fault injection experiments (described later).

The interface transforms all outputs of the vehicle model into a picture of a car moving on a road. Current state of all TTP/C nodes and their fault vectors are displayed alongside the road in small boxes, brake functionality is signalized by colored circles (green/red). The most important statistics when analyzing the braking system functionality are traveled distance and lateral deviance (deviance from a straight course). Both these statistics are displayed as current values on the main simulation screen and as a statistical variables in the statistical summary screen.

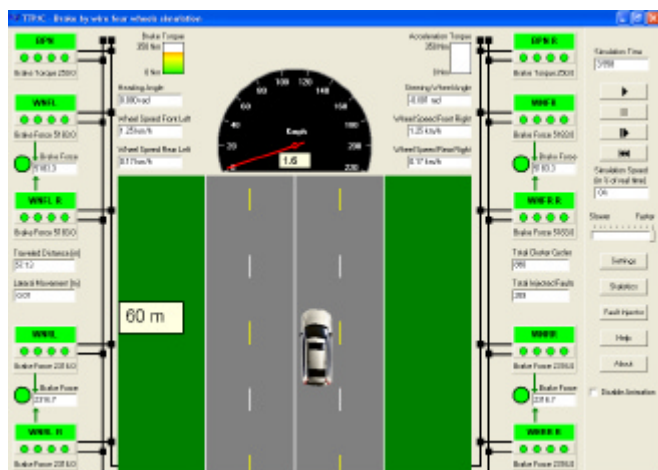


Figure 2: Visual interface screenshot

Thanks to the nature of discrete-time simulation, the execution of the model can be deliberately stopped and resumed. It is also possible to limit the speed of execution so that the behavior of the car may be studied in detail. The visual interface and the model concept are based on (Kreizek 2002).

FAULT INJECTION

Fault injection (*FI*) technique is a classic method of fault tolerance verification. It uses artificial faults induced into the verified system to check how the system copes with them. There are various FI methods, but basically we can divide them into three groups: hardware based (e.g. electromagnetic interference), software based (e.g. code and data faults) and simulation based.

For any FI experiments it is essential to have first a *fault assumption* for the tested system (promised level of fault tolerance). For a TTP/C system there are two basic assumptions (exact specification at <http://www.ttchip.com>):

- The ability to tolerate any single failure of any of its components.
- The ability to recover from short transient faults within one TDMA round.

Now it is time to build an appropriate *fault model*. A fault model can be described by several parameters, this particular model is a simplification of complicated fault hierarchy described by (Herout 2002):

- *Locality* — what part of system is attacked (e.g. memory location), how large is the fault.
- *Effect* — can be simple (e.g. set to zero, bit-flip) or quite complex (e.g. set to zero if greater then x).
- *Permanence* — if the fault can be repaired (e.g. by restarting) it is *transient*, otherwise it is *permanent*.
- *Periodicity* — the fault occurs periodically (e.g. in a Poisson stream with mean frequency λ).

The fault model should be realistic, i.e., cover all faults that are possible to occur in the tested system. This hard to accomplish using a single FI technique and so several of them in combination are used mostly. The simulation based FI technique is suitable for disturbing data elements and

data transmissions within the system, but not for disturbing the control flow. Its main advantage is that it is not limited by traditional hardware constraints, that there is no intrusion effect and that it is deterministic (i.e. reproducible).

Fault Injection into Braking Process

The aim of this paper is to prove the safety of an electronic brake control system, so we will be examining the fault tolerance of TTP/C microcontrollers and communication system, not that of a particular brake. This provides us with two possible fault injection target groups:

- Transmission channels
- TTP/C nodes

Let us examine the first case. Disruption of communication on a single channel will be safely detected by all nodes because a CRC check will fail (Hamming distance of the polynomials is 5 or 6, depending on the transmitted frame size). The second channel is in such case still operating and the fault has no impact on the operation of the nodes or of the car. If both channels fail at the same time, then all nodes that were currently sending frames will be expelled from the communication cluster and their replicas will take over their duties. The temporal distance of node replicas should be as large as possible to reduce the probability of a single fault affecting both replicas. Temporally “large” faults, that disturb the transmission of more than one half nodes in the communication cluster, will always cause a complete cluster shutdown. Such case cannot be handled by any mechanism of TTP/C and the cluster will stay inactive until the fault disappears. This type of fault injections has been already performed (on a different setup, but the results are valid for any TTP/C based application) and details can be found in (Grillinger 2002a).

The second case involves damaging selected parts of individual nodes. The fail-silence assumption has been already sufficiently proven in other experiments and published, so single node fault injections are not very interesting. However, for a safety critical application it is also important to analyze the consequences of possible multiple-node failure (such failures are always possible and quite probable in hard conditions that a car has sometimes to endure). A methodical approach to multiple node fault injections is presented in next section.

EXPERIMENT ORGANIZATION AND RESULTS

As mentioned earlier, the most important statistics of braking process are the traveled distance and deviation from course. The determining factors of these statistics are initial vehicle speed, surface adhesion, brake pedal force and the actual process of braking. Systems as ABS can improve the quality of braking process (or degrade in rare cases, but this possibility does not concern us). To make the results of all experiments comparable, all starting conditions must be set to the same values, e.g. in our case:

- Initial vehicle speed 30m/s
- Surface adhesion of dry tarmac
- Brake pedal pressure 250Nm

Different Fault Types

Experimental verification of fault tolerance at architectural level requires a deterministic fault model that can be used to cause failure of selected subsystems (here TTP/C nodes) with certainty. This allows verifying that the architecture was designed properly (e.g. node replication). TTP/C protocol contains a flag in the CNI interface called *CO*, which controls whether a node is operational or not. Single bit FI into this field produces immediate node failure, which is exactly what is required (see table 1, row 1; *Dist.* is the braking distance and *Dev.* is deviation from course, both values are in meters).

Next step in FI experiments should be verification of application level fault tolerance, i.e. the ability to handle faults in application data. In TTP/C nodes, these data are partially stored within CNI and partially in the memory of host computer. A fault in application data does not cause a failure of TTP/C communication subsystem and potentially wrong messages may be sent to other nodes (results are in table 1, row 2).

The final step is to verify that a system is able to tolerate arbitrary faults, i.e. any of its subsystems may be damaged and the system will still provide required functionality (within the limits given by fault assumption). For this purpose we inject faults into all available memory fields of TTP/C node (CNI) and host computer (see table 1, row 3).

Table 1: Different Fault Types

Fault Type	Dist. [m]	Dev. [m]
CNI – CO	93.08	2.13
Application Data	87.24	1.83
Arbitrary	61.02	0.41

Table 1 contains FI results for injection of correlated faults (term is explained later) into both replicas at front left wheel. We used a fault stream with mean period between faults equal to 5 TDMA rounds (i.e. approximately 33% of the time, the brake was malfunctioning). We can see that the CNI-CO type of fault is most effective — this is because the fault always causes temporary brake failure. The second type of fault (application data) is slightly less dangerous, because the nodes continue their operation and in some cases their data are still usable. The last type of fault is least dangerous, because there is a large number of “empty” areas in the memory of a node, and so many fault are not effective (this would change with the number of bits attacked within a single fault, now it was a single bit).

The first fault type (CNI CO field) will be used in all subsequent experiments, as it is most effective.

Reference Experiments

For comparison, we need to perform also braking without fault injection. This experiment will provide reference results that represent an optimal braking scenario (see table 2, row 1).

For the sake of completeness of our experiments we will perform also fault injection into single nodes. The results must confirm the hypothesis that the braking process tolerates this type of faults completely, i.e. the measured statistics must be exactly the same as the reference results (see table 2). The table contains one row for every node (plus the first row that contains reference no-FI results). Only results for one node of every replica-pair are presented. We used a fault stream with mean period between faults equal to 5 TDMA rounds (i.e. the same as in the experiments in previous section).

Table 2: Reference Braking Experiments

FI model	Dist. [m]	Dev. [m]
Reference run (no FI)	56.67	0.0
FL wheel (FL)	56.67	0.0
FR wheel (FR)	56.67	0.0
RL wheel (RL)	56.67	0.0
Brake pedal (BP)	56.67	0.0

Temporally Dense vs. Sparse Faults

An important parameter of fault injection is the mean frequency of incoming faults. Dense streams of transient faults can have the same effect as a permanent fault, while sparse streams of faults may not even be noticeable at higher architectural levels. Using streams with different fault frequency we will be able to determine worst-case braking statistics (for given fault model) and also to find optimal FI rate.

Table 3 summarizes results of fault injection into both replicated nodes at front right wheel (the faults are correlated, what this means is explained in the next section). We use four different fault streams — defined by mean period between successive faults. We can see that the shortest period produces the worst traveled distance (as expected), however, the lateral deviation from course is largest for the second shortest fault period. This can be explained by the function of internal braking force distribution algorithm (it is part of the vehicle model). If the brake malfunction is predictable, as is the case with almost continuous fault injection, the algorithm is able to compensate the lateral movement better than with unpredictable (from the algorithm point of view) failures. Decreasing the FI period below 1 TDMA round does not lead to significant increase of braking distance and similarly increasing the period above 20 TDMA rounds leads to results near to the optimal reference run.

For further experiments we will use mean fault injection period of 5 TDMA rounds, as the results are most promising (worst case).

Table 3: Different Fault Injection Frequency

Mean FI Period [TDMA rounds]	Dist. [m]	Dev. [m]
1	101.51	-1.83
5	92.33	-2.04
10	63.08	-0.91
20	58.56	-0.18

Multiple Node Fault Injection

The possibility that two independent faults occur within the limit of 1 TDMA round (worst recovery time for TTP/C nodes) in two different nodes is very low, it is much more probable that a fault, which occurred in two systems at approximately the same time, has the same source. What is important in such case is, whether the fault damaged both replicas in a subsystem (the subsystem fails), or nodes from different subsystems (the faults are tolerated). Common source faults (also called *correlated faults*) become probable if the architecture of a system uses shared resources for multiple nodes (e.g. single power supply). Another common source of correlated faults are microcontroller programming bugs. The ability of a simple TTP/C cluster to recover from multiple node failures was examined in (Grillinger 2002b).

Table 4 summarizes the results of correlated fault injection into different node combinations. The first row experiment damaged both brake pedal nodes, the second both front left wheel nodes, the third two different wheel nodes and the fourth one wheel node and one brake pedal node. To see the difference between correlated and random multiple faults, the table contains also results for random FI (lower half of the table).

Table 4: Multiple Node Fault Injection

FI Model	Target Nodes	Dist. [m]	Dev. [m]
Correlated	BP1 + BP2	858.72	0.00
	FL1 + FL2	93.08	2.13
	FL1 + RL1	0.00	0.00
	BP1 + RR1	0.00	0.00
Random	BP1 + BP2	289.05	0.00
	FL1 + FL2	66.77	-1.17
	FL1 + RL1	0.00	0.00
	BP1 + RR1	0.00	0.00

The results in table 4 show that the braking process is disturbed only when both replicas are damaged. This confirms the fail-silence hypothesis of TTP/C and also implies that correlated faults have greater potential to cause brake failures (correlated faults always damage multiple nodes at the same time, independent faults have small probability of affecting multiple nodes). The danger of correlated faults is obvious when cross-checking the results in table 4.

CONCLUSIONS

The scope of this paper does not permit to completely cover the problem of verification of a complex system as braking-by-wire. The presented methods are sufficient to explore most of possible failure scenarios and the selected

results give a basic estimation of safety achieved in the modeled system. Although the model is capable of detailed simulation of TTP/C protocol, low-level analysis of its behavior is not the aim of this work. The focus is on the safety at application level and correct behavior of the whole system.

It was demonstrated that correlated faults are significantly more dangerous than independent (random) faults. Fault correlation is mostly caused by sharing resources between replicated nodes and/or spatial proximity of replicas — a practice that can and should be avoided at design phase. Experiments with different fault injection frequencies are useful for worst case analysis (finding as well as evaluating the worst FI frequency).

The visual interface is a very useful demonstration tool as it allows anyone without deeper knowledge of TTP/C or the vehicle model to perform own experiments and to see the results without the need to read through number of different statistics. It provides clear answers to questions like “What will happen if the wires to a wheel are cut?”

This paper considers only a car that is heading straight ahead and uses only the most basic faults. A future model of brake-by-wire should enable experiments under different conditions, e.g. a car driving through a road bend.

REFERENCES

- Grillinger P. and Racek S. 2002a. “Simulation-based evaluation of TTP/C controller reintegration time.” In *Proceedings of Applied Electronics Conference*, (Plzen, Czech Rep., Sep 2002).
- Grillinger P. and Racek S. 2002b. “Transient faults robustness evaluation of safety critical system using simulation.” In *Proceedings of Baltic Electronics Conference*, (Tallinn, Estonia, Oct. 2002).
- Herout P.; Racek S.; and Hlavicka J. 2002. “Model-based dependability evaluation method for TTP/C based systems.” In *Lecture notes in Computer Science no. 2485*, (EDCC-4, Toulouse, France, Oct), Springer-Verlag, ISBN 3-540-00012-7.
- Kopetz H. 1997. *Real-Time Systems, Design Principles for Distributed Embedded Applications*. Kluwer Academic Publishers.
- Krejcek T. 2002. *Verification of Application Reliability in TTA*. MSc. Thesis at Czech Technical University in Prague, Czech Republic.
- Lennevi J. 2001. “Simple vehicle model with ABS control for the FIT project.” Volvo Technological Development Memorandum 01-01-11.
- Lönn H. 2001. “Brake by Wire - Status Report”. EU FIT project status report, Prague, Czech Republic.
- Manzone A. et al. 2001. “Fault tolerant automotive systems: An overview.” In *Proceedings of 7th Int'l On-Line Testing Workshop*. (Taormina, Italy, July 2001). 117-121.

ACKNOWLEDGEMENT

The research was in part supported by a grant of the Grant Agency of the Czech Republic — Research of methods and tools for verification of embedded computer systems, no.102/03/0672. Theoretical part of work was supported by the Ministry of Education of the Czech Republic, project no. MSM-235200005 — Information systems and Technologies.

Computer Simulation of Seat Vibration Test and Design Optimization of a Vehicle Suspension Seat Using an Equivalent 1-D.O.F. Mass-Spring-Damper Model

Hoon-Seok Byun, Sung-Nam Heo
Graduate Student of Mechatronics Engineering Department
Changwon National University
9 Sarim Changwon, Gyeongnam, 641-773, Korea
E-mail: manilkka@hotmail.com

Young-Hyu Choi, Joong-Ho Shin, Soo-Tae Kim
Department of Mechatronics Engineering
Changwon National University
9 Sarim Changwon, Gyeongnam, 641-773, Korea
E-mail: yhchoi@changwon.ac.kr
E-mail: jhshin@sarim.changwon.ac.kr
E-mail: stkim@sarim.changwon.ac.kr

KEYWORDS

Parallel Suspension Seat, Seat Vibration Test, Computer Simulation, Design Optimization, Genetic Algorithm

ABSTRACT

This paper presents a case study on the computer simulation of a vehicle suspension seat vibration test, whose method is specified in the ISO's test codes. Firstly, a real driver-suspension seat system was modeled as an equivalent 1-D.O.F. mass-spring-damper system. And vertical vibration response and transmissibility due to base excitations were computed. In order to investigate the validity of the computer simulation, the damping test for suspension seat system was carried out. For the suspension dynamic design optimization, a simple genetic algorithm was adopted to searching the optimum values of the design variables.

1. INTRODUCTION

The trend of computer simulation based design is recently increasing in the field of industrial machine design. This paper presents a case study on the computer simulation of seat vibration test and design optimization of a suspension seat system for an off-road vehicle. Firstly, a real driver-suspension-seat-system was mathematically modeled as a 1 DOF mass-spring-damper system. A computer simulation program was developed in order to perform the seat vibration test, in virtual, which is specified on ISO 7096 and 10236-1 test codes. In order to verify the developed computer simulation program, some experimental seat vibration tests were carried out. Both experimental and computer simulation based seat vibration test results show good agreement. Also the program was successfully applied to optimizing the suspension parameters such as spring constant and damping constant. For that purpose, it was combined with a design optimization program using a simple genetic algorithm.

2. VIBRATION ANALYSIS OF A DRIVER-SUSPENSION-SEAT SYSTEM

2.1 Modeling

A driver-suspension-seat-system under base excitation was mathematically modeled as a linear 1 DOF vibrating system as shown in the Figure 1.

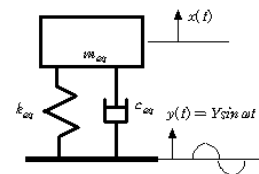


Figure 1 : An Equivalent 1 DOF Modeling of A Driver-Suspension-Seat-System Under Base Excitation

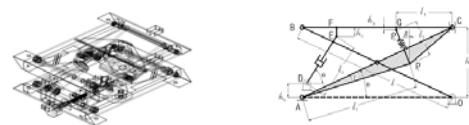
In this model, only vertical motion was considered because the others were assumed very small. The equation of motion for the equivalent model is derived as following by using Newton's 2nd law:

$$m_{eq} \ddot{x} + c_{eq} \dot{x} + k_{eq} x = k_{eq} Y \sin \omega t + c_{eq} \omega Y \cos \omega t \quad (1)$$

Where m_{eq} , c_{eq} , and k_{eq} are equivalent sprung mass, equivalent damping constant, and equivalent spring constant of the driver-suspension-seat system respectively.

2.2 Equivalent spring and damping constants.

A kinematic model as shown in the Figure 2 (b) was made in order to calculate the equivalent suspension parameters for the real pneumatic suspension mechanism shown in Figure 2 (a).



(a) Real Suspension

(b) Kinematic Model

Figure 2 : A Kinematic Model of a Pneumatic Suspension

The equivalent spring constant k_{eq} can be obtained as follows by the kinematic relations of the parallel suspension mechanism:

$$k_{eq} = k_{ps} \cos \beta \quad (2)$$

Where, k_{ps} is the spring constant of the spring in the real parallel suspension mechanism. And β is the spring's setting angle about vertical axis.

Similarly the equivalent damping constant c_{eq} can be obtained as follows:

$$c_{eq} = c_{sa} \sin \alpha \quad (3)$$

Where, c_{sa} is the damping constant of the parallel suspension mechanism. And α is the ? angle about vertical axis.

2.3 Solution

The vibration response of the equivalent driver-suspension-seat model due to a base excitation can be obtained as follows by solving the equation (2):

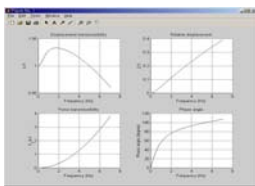
$$x(t) = e^{-\xi \omega_d t} \left\{ x_0 \cos \omega_d t + \frac{\dot{x}_0 + \xi \omega_n x_0}{\omega_d} \sin \omega_d t \right\} + \frac{Y \sin(\omega t - \phi_1)}{\sqrt{(1-r^2)^2 + (2\xi r)^2}} + \frac{2\xi r Y \cos(\omega t - \phi_1)}{\sqrt{(1-r^2)^2 + (2\xi r)^2}} \quad (4)$$

Where, ξ is damping ratio, x_0 is the initial displacement, \dot{x}_0 is the initial velocity, and ω is exciting frequency, and Y is the amplitude of the base excitation respectively. In equation (4), $T(f_r)$, the maximum acceleration transmissibility of the equivalent suspension seat model, can get as following equation.

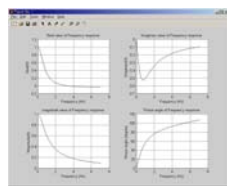
$$T(f_r) = \left\{ \frac{a_s(f_r)}{a_p(f_r)} \right\} = \frac{\sqrt{1 + 4\xi^2}}{2\xi} \quad (5)$$

3. SIMULATION OF SEAT VIBRATION TEST

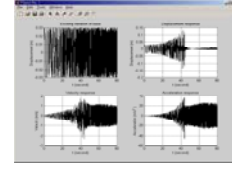
ISO 7096 and 10326-1 suggest two kinds of test, that is, Simulated Input Vibration test (SIV test) and Damping test. First, SIV test defines four spectral classes of input vibration and each class is defined by a normalized power spectral density (PSD). Figure 3 shows some typical examples of the simulation results of SIV test using the our-own made simulation program. Second, damping test should be loaded with an inertia mass of 75kg and then be excited by a sinusoidal vibration in the range from 0.5Hz to 3Hz. Damping test also shows same kind of graphs.



(a) transmissibilities



(b) frequency responses



(c) time history responses

Figure 3 : Some Typical Examples of Computer Simulation Results of the SIV test

Computer simulated acceleration transmissibility result will be compared with measured test result in the following section

4. VERIFICATION

Although ISO suggests two kinds of seat vibration test, damping vibration test only was executed, in this paper..

4.1 Damping vibration test

The test parameter setup for the damping vibration test is shown in Table 1.

Table 1 : Test Parameter Setup For The Damping Test of a Driver-Suspension Seat System.

Parameter		Value
Parallel Link Length, L		320 mm
Equivalent Sprung mass, m_{eq}	Driver mass, m_{lp}	75 kg
	Seat Sprung mass, m_{seat}	2 kg
Frequency Sweep		0.5 Hz ~ 4.5 Hz
Frequency Increment, Δf		0.05 Hz
Excitation Amplitude		0.030 m (peak to peak)

Spring constant of the pneumatic suspension, k_{ps} and damping coefficient of the shock absorber, c_{sa} for each pressure condition were shown in Table 2.

Table 2 : Measured Equivalent Spring Constant and Damping Coefficient of the Suspension Seat System

Air spring Pressure [bar]	k_{ps} [N/m]	c_{sa} [N·s/m]
3.0	69800	1061
3.5	61640	1050
4.0	47086	1040

4.2 Damping vibration test results

Table 3 and Figure 4 present the comparison of the both measured and theoretical acceleration transmissibilities of the seat suspension system.

Table 3 : Comparison of measured vibration transmissibilities of Equivalent Spring Constant and Damping Coefficient of the Suspension Seat System

	Air spring Pressure (bar)	$a_p(f_r)$ [m/s ²]	$a_s(f_r)$ [m/s ²]	f_r [Hz]	$T(f_r)$
Expriment	3.0	8.60	14.1	3.9	1.69

	3.5	4.24	6.32	3.5	1.49
	4.0	3.08	4.08	2.7	1.33

As shown Table 3 and Figure 4, both real test and computer simulation results are in good agreement.

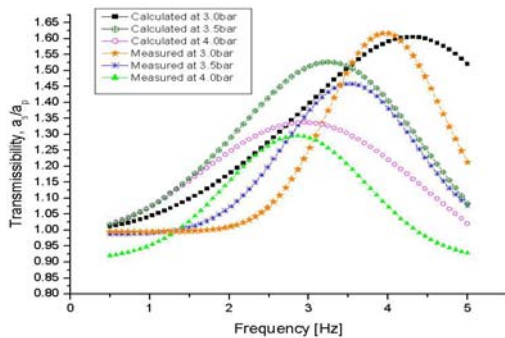


Figure 4 : Comparison of Measured and Computed Acceleration Transmissibilities of the Suspension Seat

5. APPLICATION IN THE SEAT SUSPENSION DESIGN OPTIMIZATION

5.1 Design variable and objective function

The equivalent 1-D.O.F. driver-suspension-seat model and computer simulation program for seat vibration test can also be applied to the dynamic design optimization of seat suspension parameters. In this paper, the dynamic design optimization problem of a suspension seat system is defined as follows.

$$\text{Minimize } f(k_{ps}, c_{sa}) = \text{Min} \left\{ \begin{matrix} a_s(f_r) \\ a_p(f_r) \end{matrix} \right\} \quad (6)$$

$$\text{Subject to } g_1 = |X-Y| < d_1, \quad G_2 = T(fr) < T_c$$

Where, d_1 is rattle space limit

T_c is set vibration trasmissibility

Design variables in this case are set as the air spring constant, k_{ps} and the damping constant of the shock absorber, c_{sa} .

A simple genetic algorithm was used in order to search the otimum solution. The input parameters for , used in genetic algorithm, are shown in Table 4.

Table 4 : Parameters for Optimization by G.A.

Parameters	Value	Parameters	Value
Rattle space limit d_1	114mm	Population Size	30
		Crossover rate	0.9
Transmissibility limit, T_c	1.5	Mutation rate	0.001
Maximum generation	100	Length of binary string	37

5.2 The result of optimization.

As the result, optimum values of the design variables are finally found as air spring constant k_{ps} about 38000N/m and and shock absorber damping c_{sa} about 8000 N·s/m

respectively. Figure 5 shows comparison of calculated and measured vibration transmissibilities for the optimized suspension seat system.

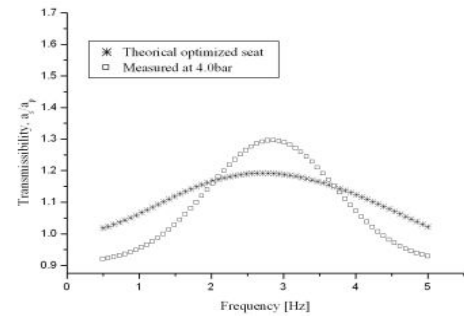


Figure 5 : Comparison of Transmissibilities for the Optimized Suspension Seat System

From the Figures 4 and 5, It is apparently seen that the acceleration transmissibility level of the optimized seat is decreased by about 1/4 compared with non-optimized seat parameters. Therefore, this program can be successfully applied to dynamic optimization of suspension seat system for reduced vibration level.

6. CONCLUSION

In this paper, a computer simulation of the ISO's seat vibration test and design optimization of a suspension seat system were studied. Firstly, a driver-suspension-seat-system was mathematically modeled as a 1 DOF mass-spring-damper system. Then a computer simulation program was developed in order to perform the seat vibration test in virtual. Both results from computer simulation and actual seat vibration test show good agreement. Furthermore, the simulation program, after being combined with a G.A. based design optimization program, was successfully applied to dynamic design optimization of the suspension parameters such as spring and damping constants.

REFERENCES

- International Organization for Standardization, 1994, Earth-Moving Machinery - Laboratory Evaluation of Operator Seat Vibration. ISO 7096.
- International Organization for Standardization, 1992, Mechanical Vibration – Laboratory Method for Evaluating Vehicle Seat Vibrtn. ISO 10326-1.
- Singiresu S. Rao. 1992. Mechanical Vibrations. 2nd ed., Addison Wesley. Inc.
- Y.H. Choi, H.S. Byun, and H.I. Jung, 1999, "Vibration Analysis and Test of a Suspension Seat Using Computer Simulation," Proc. Of KSME '99 autum conference, Vol. A, pp. 916-921.
- Y.H. Choi, H.S. Byun, and H.I. Jung. 1999. "Dynamic Design Optimization of a Pneumatic Suspension Seat Using Genetic Algorithm," Proc. of KSPE '99 Autum conference, pp.745-748.

Simulation in Automotive Design

CAD-BASED ERGONOMICS ANALYSES IN MOTORCYCLE DESIGN

Sandro Barone and Alessandro Curcio
Dipartimento di Ingegneria Meccanica, Nucleare e della Produzione
University of Pisa
Via Diotisalvi 2, 56126 Pisa,
Italy
E-mail: s.barone@ing.unipi.it, alessandro.curcio@ing.unipi.it

KEYWORDS

Design for ergonomics, CAD, Motorcycle Design.

ABSTRACT

In this paper, a CAD-based tool for ergonomics analyses of motorcycles is presented. The system can automatically prognosticate driver's postures with male and female body types subdivided in age, nationality and percentage groups. Driver digital models are assembled into virtual vehicle prototypes, and natural postures are obtained and rated on the basis of specific behavioural criteria.

The system is integrated into a top-down framework for motorcycle design. As a result, vehicle designers are able to use a single analysis tool to assess clearance, reach and postural comfort at the earliest stages of the design process. In the paper, issues in the analysis of vehicle designs regarding the use of supporting ergonomics systems, assumptions and posture evaluations are discussed.

INTRODUCTION

During the last decades, several anthropometrics and ergonomics studies have been conducted for the design of workplaces and industrial products [1-2]. For example, ground vehicle industry is one of the sector which has strongly faced up to the ergonomics challenge [3]. However, whereas automotive industry has positively responded to ergonomics requirements through experimental activities and development of computerised analysis tools, motorcycle design is still characterised by insufficient ergonomics considerations [4-5].

Traditionally, ergonomics analysis in motorcycle industry has been limited to two-dimensional simulations and few experimental tests. A study of the ergonomics analysis procedures adopted in motorcycle design has evidenced the following deficiencies:

- weak integration between different analysis steps;
- limited use of automated procedures;
- evaluations based on subjective judgments.

These deficiencies indicate the need of ergonomics analysis procedures which should be automated, integrated to the design process, based on reliable anthropometric data, and usable since the preliminary vehicle design phases.

In this paper, the development of a 3D CAD-based ergonomics analysis system to support motorcycle design is

described. This research activity has been conducted in collaboration with Piaggio S.p.A., a leader company in motor-scooter design and manufacture.

The ergonomics analysis system is based on integrating an anthropometric digital model to a motorcycle design framework, developed in a previous research activity [6]. Postures coherent with human physiology are obtained and rated in accordance to specific behavioural criteria defined through experimental tests.

It has been noted that the quality of decision-making can improve enormously if postural parameters of a vehicle can be expressed as numbers. This approach allows concrete comparisons with competing or alternative motorcycle solutions and avoids mistakes caused by subjective judgements. Vehicle designers can change a geometric configuration and predict the driver's posture at development stage without any need for physical mock-up and tests. Moreover, the system enables vehicle layouts to be adjusted on the basis of the anthropometric peculiarities of customer groups.

A HUMAN CAD MODEL

A CAD-based manikin has been defined to carry out ergonomics static analysis of motorcycles by rating postures with various typologies of drivers and vehicle configurations.

The manikin is composed of a skeleton with assembled external body parts. In particular, the human bone structure has been schematised by considering the following joints (Figure 1-a): wrist (joint 4-5), elbow (joint 3-4), knee (joint 6-7), shoulder joint (joint 4-5), ankle, backbone joints at first and second cervical vertebrae, lumbar-sacrum joint (joint 0-1), sacrum-iliac joint (joint 0-6). The bone structure scheme has been represented, using a top-down solid modelling approach, as a skeleton (Figure 1-b) that contains datum coordinate systems to assemble the parametric body parts (e.g., arms, legs, hands, feet, bust, head, joints) and reference points (e.g., left and right hands, left and right feet, and seat references) to interface to a vehicle model.

Skeleton and body parts of the manikin have been modelled using family tables in order to include dimensional variants on the basis of an anthropometric database. In particular, potential motorcycle users have been gathered, as described later on, into age, nationality, gender and percentile groups.

The designer selects a user group and a virtual manikin (Figure 1-c) with the specified anthropometric properties, is automatically assembled into a 3D vehicle model.

For this purpose, the human model has been included, as additional functionality, into a vehicle design framework based on a top-design approach [7]. This system enables motorcycle layouts to be defined on the basis of parametric layouts that perform the desired functions, meet the performance standards, and satisfy constraints. Vehicle layouts are adaptable to different dimensions and mechanical solutions, and represent the basic structures for the development of virtual prototypes of vehicles using feature-based assembly procedures (Figure 2-a). Moreover, the design framework is able to represent a physical vehicle prototype as a 3D sketch (Figure 2-b) including space lines and vehicle-driver references, which have been experimentally measured by a control measure machine and reconstructed within the CAD environment. This procedure enables posture evaluation of competitor vehicles and physical mock-ups to be carry out using the CAD-based ergonomics analysis system.

The virtual manikin is automatically assembled into a 3D vehicle model, defined either as an assembly of solid parts or as a 3D sketch of ergonomics surveys, by aligning the interface points. In particular, datum points are used to align hands, feet and seat to handlebar grips, footboard and saddle, respectively. In real time, the system provides the joint angles assumed by the virtual driver belonging to the specified user group, and compare them with optimal ranges, experimentally evaluated, to have an immediate indication of postural comfort.

However, the driver-vehicle interface points do not provide a univocal posture, since the manikin is not completely constrained. Behavioural criteria of postural comfort of motorcycles have been defined through an experimental activity. In particular, a behavioural model provides comfortable joint angle ranges, which are used as comparison for posture evaluation and to assign the values to the joint angles that still remains undefined as the manikin is assembled to a vehicle model. The manikin assumes the most natural postures with the unknown joint angles defined on the basis of driver anthropometry and vehicle geometric configuration. Later on, more details concerning the development of postural criteria are given.

In addition, the ergonomics analysis system incorporates basic range of motion constraints on the major joints to avoid unlikely or impossible postures. For example, the knee and elbow joints are restricted to a range 0° ÷ 180° , representing full flexion to full extension, respectively. Figure 3 shows an example of manikin simulating a driving posture along with the graphical interface to select the ergonomics analysis parameters.

The system enables the ergonomics analysis of different vehicle design solutions by simulating and rating the postural behaviour of different users' groups. The vehicle layout can be varied by manual displacement of the interface points: the ergonomics analysis system will congruently adjust the manikin's posture. Moreover, postures assumed for some manoeuvres of wheel turning (Figure 4), such as occurring parking the vehicle or driving at low speed in heavy traffic, can be analysed.

The results of simulation tests can be verified by experimental comparisons using a vehicle mock-up (Figure 5) whose configuration can be easily adjusted on the basis of the geometrical parameters provided by the vehicle design framework. Consequently, the driver's posture can also be experimentally tested, before a vehicle prototype is manufactured.

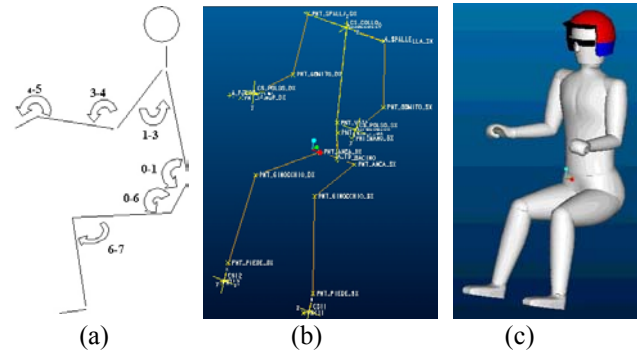


Figure 1. Virtual manikin: scheme with joint angles (a), skeleton representation (b), solid model (c).

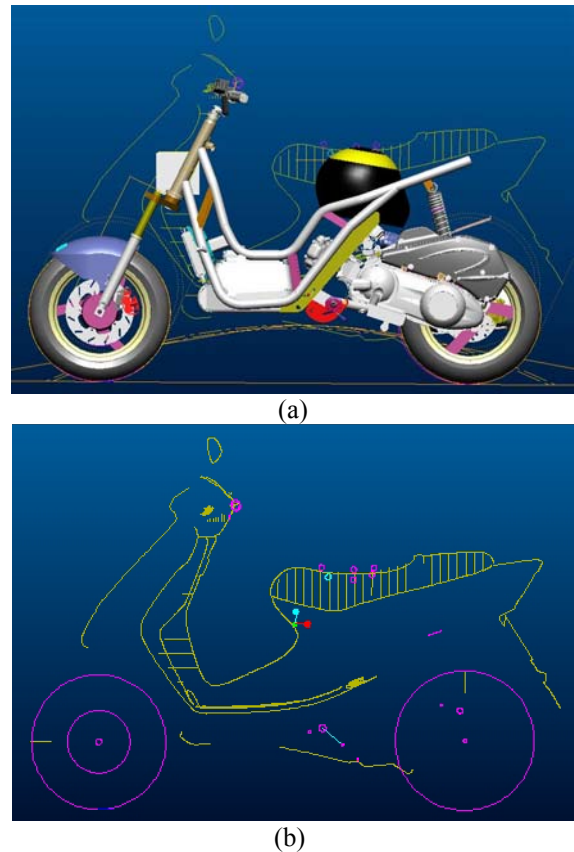


Figure 2. CAD model (a) and ergonomics surveys (b) of motor-scooters.

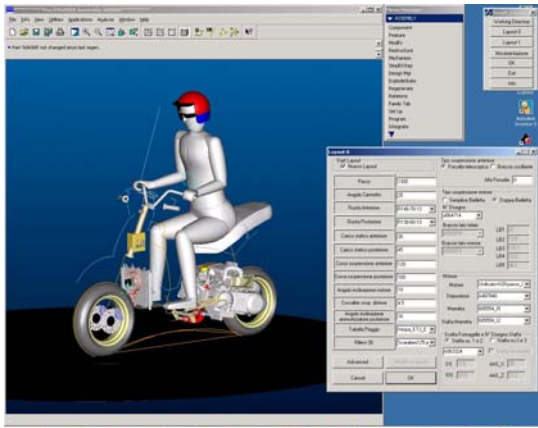


Figure 3. Ergonomics analysis using a 3D manikin linked to a vehicle model.



Figure 4. Posture analysis simulating different vehicle manoeuvres.



Figure 5. Vehicle mock-up for posture analyses.

AN ANTHROPOMETRIC DATABASE

The vehicle design can be affected by necessities of optimising driving configurations, shapes, adjustments and geometries of driver-vehicle interfaces (saddle, handlebar and footboard) on the basis of anthropometric peculiarities of potential users. Thus, an anthropometric database has been defined in order to represent customers who should fit particular vehicle configurations.

The database for the human CAD model has been created by using the commercial program PeopleSize™ 2000 by Open Ergonomics Ltd, which provides data about static anthropometrics, expressed as percentiles of populations, where measurements are of people in fixed postures. The anthropometric data, which have been gathered from a

large number of sources [7-9], are statistically processed and grouped on the basis of ages, genders and nationalities. The dimensional data obtained from PeopleSize™ 2000 have been used to create the family tables of the body parts within the CAD environment. In this work, 54 anthropometric populations have been created and subdivided on the basis of ages, percentiles, nationalities and gender. In particular, three percentiles (5, 50, 95) have been associated to gender (male, female), three age ranges (14_17, 18_25, 25_50), and three nationalities (European Union, Japan, United States) in order to obtain 54 variants for each body part.

Only body part dimensions have been extracted from PeopleSize™ 2000, without considering posture parameters, since, typically, the measurements are presented in static positioning, which are not referred to any vehicle configuration. Thus, posture tests in realistic situations have been conducted. This is also advised because the rigid nature of static anthropometrics does not always match the dynamic movements of a driver. Motorcycle drivers move dynamically and assume specific postures on the basis of vehicle typology and driving experience. In this work, anthropometric data concerning sizing of driver populations have been correlated to posture criteria, experimentally obtained for various vehicle configurations.

POSTURE MODELLING

The development of a CAD-based ergonomics analysis system has pointed out the necessity of a motorcycle driving posture model, which provides indications of joint angles associated to comfort perception. This has been obtained by correlating experimentally measured joint angles with judgements expressed by test drivers. In this activity, few test drivers have been used, whereas, a more complete analysis would require experimental tests with drivers presenting different anthropometric peculiarities.

In the experimental tests, driver's joint angles have been measured using an accurate portable measurement arm (FARO arm) with a mechanical probe that provides 3D coordinates of targets (metallic plates) attached to the driver body parts (Figure 6). The probe is composed of a cylindrical body and a quadrilateral mouthpiece, which is aligned to a target groove. The vehicle is positioned on a test bench and blocked through pins on the centre wheels and jack arms (Figure 7). A calibrating table composed of targets with known relative positions, is positioned close to the base in order to optimise the working distance of the FARO Arm.

Each target is associated to a local coordinate system whose position and orientation is univocally determined with reference to the global coordinate system defined by the calibrating table. A calibration process is preliminarily carried out by measuring positions of markers on the calibrating plate and on the test driver standing. Then, the test driver assumes the driving position and the posture is measured. The experimental data obtained by the FARO Arm are then processed to calculate rotations and translations of each local coordinate system to pass from

the calibration position (stand posture) to the driving position. The posture parameters are directly given in terms of joint angles assumed by the test driver (Figure 1-a): wrist (joint 4-5), elbow (joint 3-4), knee (joint 6-7), shoulder joint (joint 4-5), ankle, backbone joints at first and second cervical vertebrae, lumbar-sacrum joint (joint 0-1), sacrum-iliac joint (joint 0-6).

The experimental data have been associated to subjective perceptions of postural comfort provided by the test drivers by rating (points from 6 to 10) five ergonomic parameters:

- geometrical habitability with stationary vehicle;
- physiological habitability on the basis of postures assumed in static and dynamic conditions.
- accessibility of principal commands;
- protective capabilities from atmospheric elements;
- ergonomics and functionality of accessories.

Of course, there is a relationship between some of the elements above indicated. However, in this work, the behavioural model has been defined by processing data only concerning physiological habitability, which mostly influences the evaluation of postural comfort of a vehicle.

Drivers have had to judge the postural comfort in static conditions (five minute with stationary vehicle) and dynamic conditions (road tests for over 40 minutes). For each vehicle, the drivers could percept pains and disturbs due to: *i*) relative positions and orientations of handlebar, footboard and saddle; *ii*) vibrations; *iii*) weight and cyclist of the motorcycle, particularly at slow speeds. Some discrepancies could be evidenced analysing static and dynamic results. For example, some vehicles with a good score in static conditions, have received a poor score in dynamic tests due to vibration phenomena. However, it is almost impossible to weight each element in a global score. Thus, for the analysis conducted in the work, only vehicles with positive scores (points 6 to 10) both in static and dynamic test conditions have been considered for further processing.

Eighteen different motorcycles covering a wide range of motor-scooters have been tested by three test drivers. Figure 8 shows the plots of joint angles versus vehicle code numbers ordered with increasing comfort scores. The results are referred to correlating the average of five different measurements with the rating expressed by a test driver. The figure 8 also shows four score zones, from 6 to 9. The experimental results indicate that, generally, the variations of joint angles decrease with increasing the static comfort. Independent ranges of joint angles can be also obtained grouping the experimental results for specific scores. Tables 1, 2 and 3 report the joint angle ranges corresponding to point 6, points 8-9 and point 9, respectively. Joint angles associated to high scores provide confident values of a good comfort. These data have been included into the ergonomics analysis system to provide a quantitative comparison between the manikin's posture and ranges of comfortable joint angles.

As previously anticipated, the reference points on the vehicle model do not enable the manikin to assume a univocal posture, in that it is not completely constrained. In particular, two joint angles have to be defined by external assumptions. In this paper, behavioural criteria have been developed relating experimental joint angles with vehicle

parameters and driver anthropometric data. These parameters include the distances between some vehicle parts, such as footboard-saddle, handlebar-saddle and saddle-ground.

Firstly, the relations between consecutive joint angles have been found. Figure 9 shows the plot of joint angles (experimentally measured) for consecutive links: 1-0-6 (Figure 9-a) and 5-4-3 (Figure 9-b). Then, the joint angles have been associated, in a reference table, to test driver anthropometric groups and vehicle configurations. The procedure to assemble a driver model into a vehicle prototype proceeds as follows:

1. input the geometrical parameters which define the vehicle configuration (coordinates of driver-vehicle interface points);
2. define the joint angle 6-7 (Figure 1-a) from the relative positions of footboard and saddle;
3. define initial values of the joint angles 0-6 and 4-5 on the basis of vehicle configuration and test driver parameters (joint angles are extracted from the reference table choosing the values corresponding to the driver anthropometric group and to the most similar vehicle configuration among those experimentally tested);
4. univocally evaluate the unknown joints using the regression data reported in Figures 9.
5. optimise the joint angle values on the basis of cost functions that minimise the levels of discomfort from the most neutral positions of the joints [10] (a neutral position is defined as the mean value of a joint angle range corresponding to high score).

This procedure provides the posture that a test driver representing a user group, would assume with the given vehicle configuration. Ergonomics CAD-based simulations conducted with vehicles very similar to those experimentally tested in this work and with manikins reproducing the test drivers, have provided results that agree well with the experimental indications. Discrepancies between simulator responses and postural perceptions depend on the difference between the physical vehicle configuration and the most similar vehicle extracted from the reference table. On the contrary, driver antropometric parameters can be reliably reproduced. A complete functionality of the behavioural model needs more experimental tests conducted varying vehicle configurations and test drivers, to cover all the possible vehicle-user interactions.

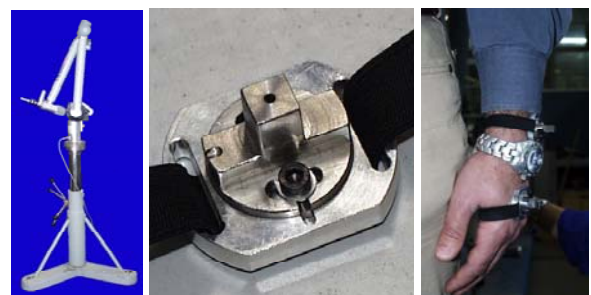


Figure 6. Experimental devices, from left to right: Faro-arm, target, targets on test driver.



Figure 7. Experimental set-up for measurement of motorcycle posture.

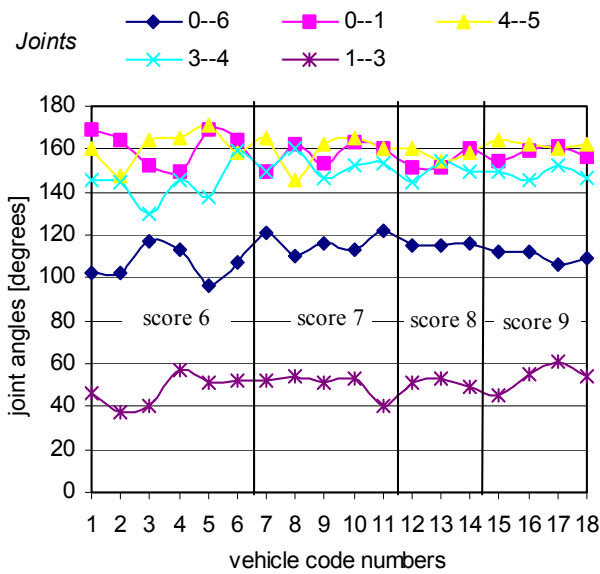


Figure 8. Plots of joint angles (in degrees) versus vehicles (indicated with code numbers) ordered with increasing postural comfort scores by a test driver.

Table 1. Minimum and maximum values of joint angles equivalent to rating 6.

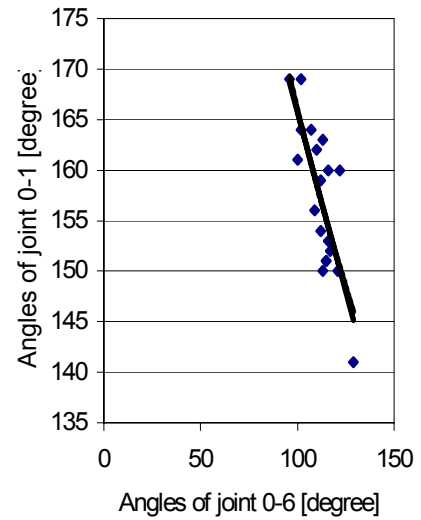
Joints	Min values [Degree]	Max values [Degree]
0-6	96	122
0-1	150	169
1-3	37	61
4-5	146	171
3-4	130	160

Table 2. Minimum and maximum values of joint angles equivalent to rating 8-9.

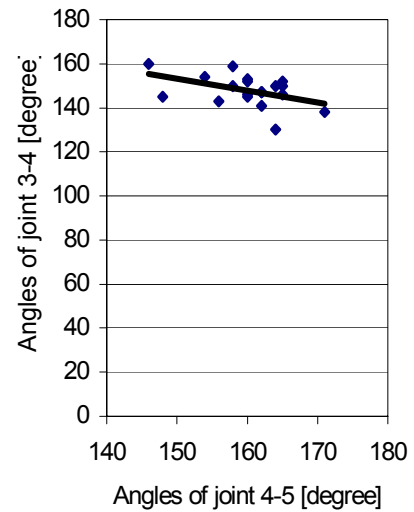
Joints	Min values [Degree]	Max values [Degree]
0-6	100	116
0-1	151	161
1-3	49	61
4-5	154	162
3-4	141	154

Table 3. Minimum and maximum values of joint angles equivalent to rating 9.

Joints	Min values [Degree]	Max values [Degree]
0-6	100	112
0-1	156	161
1-3	54	61
4-5	160	162
3-4	141	152



(a)



(b)

Figure 9. Plots of joint angles for consecutive links: links 1-0-6 (a) and links 5-4-3 (b).

CONCLUSIONS

In this paper, a CAD-based system for posture analysis of motorcycles has been described. The development of the ergonomics analysis system has been supported by the definition of an anthropometric data-base, which provides users' groups on the basis of percentile, age, nationality and gender, and by the development of a behavioural model defined correlating experimental measurements with postural perceptions provided by test drivers.

The main functionalities of the ergonomics analysis system are:

- ❑ natural postures are obtained in accordance with specific behavioural criteria;
- ❑ postural analyses of motorcycles can be conducted since the early design phases by quickly analysing the effects of positions and orientations of driver-vehicle interfaces (handlebar, footboard, saddle);
- ❑ check of current designs against different user groups (easily accessible from the anthropometric database) can be performed ensuring new designs fit the desired percentage of population.

This work has been based on a relatively small sample and clearly further work is required to obtain more detailed data on sitting position, particularly with respect to correlations between vehicle configurations and anthropometric peculiarities.

REFERENCES

- [1] A.K. Sengupta, Biman Das, "Human: an Autocad three dimensional anthropometric human model for workstation design", *International Journal of Industrial Ergonomics*, Vol.19, 1997, pp.345-352.
- [2] T. Seitz, J. Balzulat, H. Bubb, "Anthropometry and measurement of posture and motion", *International Journal of Industrial Ergonomics*, Vol.25, 2000, pp.447-453.
- [3] R. Feyen, Y. Liu, D. Chaffin, G. Jimmerson, B. Joseph, "Computer-aided ergonomics: a case study of incorporating ergonomics analyses into workplace design", *Applied Ergonomics*, Vol.31, 2000, pp.291-300.
- [4] S. Robertson, J.M. Porter, "Motorcycle ergonomics: an exploration study", in Megaw E.D. (ed) *Contemporary Ergonomics*, 1987, Taylor and Francis, pp.173-178.
- [5] S. Robertson, A. Minter, "A study of some anthropometric characteristics of motorcycle riders", *Applied Ergonomics*, Vol.27, No.4, 1996, pp.223-229.
- [6] P. Aspettati, S. Barone, A. Curcio, M. Picone, "Parametric and feature-based assembly in motorcycle design: from preliminary development to detail definition", *CD Proceedings of the 12th International Conference on Design Tools and Methods in Industrial Engineering*, Rimini, 2001.
- [7] C.M. Haslegrave, "Anthropometric profile of the British car driver", *Ergonomics*, Vol.23, No.5, 1980, pp.437-467.
- [8] K.H.E. Kroemer, "Engineering Anthropometry", *Ergonomics*, Vol.32, No.7, 1989, pp.767-784.
- [9] S.T. Pheasant, "Bodyspace, Anthropometry, Ergonomics and Design", Taylor & Francis, 1988.
- [10] K. Abdel-Malek, Wey Yu, Zan Mi, E. Tanbour, M. Jaber, "Posture prediction versus inverse kinematics", *Proceedings of ASME 2001 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Pittsburgh, 2001.

NON-STRUCTURED SIMULATION ENVIRONMENT FOR COLLISION DETECTION AND OBSTACLE AVOIDANCE ALGORITHM TESTING

Marta C. Mora and Josep Tornero
Department of Control and Systems Engineering
Polytechnic University of Valencia
Camino de Vera s/n, 46022 Valencia
Spain
E-mail: marmoag@isa.upv.es , jtornero@isa.upv.es

KEYWORDS

Simulation, Collision Detection, Obstacle Avoidance, Autonomous Vehicle, Teleoperation.

ABSTRACT

The present paper offers a thorough description of a 3D non-structured simulation environment made for testing collision detection and obstacle avoidance algorithms between an autonomous or teleoperated vehicle and random mobile obstacles on a highway environment. The autonomous vehicle tries to follow a prestablished trajectory while the teleoperated vehicle is driven by the user in real-time.

A commercial programming software called DarkBasic® is used for modeling vehicles and environment.

Control algorithms are implemented as DLL functions exchanging data with the simulation environment and with the real application if required.

Specifically, it has been applied to test a collision detection and obstacle avoidance algorithm based on the computation of the Minimum Translational Distance between spheres.

INTRODUCTION

Nowadays, virtual environments are widely used for testing algorithms in mobile robotics. It is particular useful when real implementation would be dangerous. In such situations, it is essential to detect any possible fault that may exist in the algorithm. In addition, random phenomena can be generated and their consequences evaluated. In spite of the fact that simulation does not contemplate every factor involved in real situations, it is still a good approximation. Simulations also allow a better understanding of system behavior (Barton 2001).

The paper gives an accurate description of a 3D simulation environment created specifically to test collision detection and obstacle avoidance algorithms. It is termed a non-structured environment in the sense that there are not regular references in the environment to be taken into account for autolocalization. There is no *a priori* knowledge about the obstacles in the vehicle surroundings.

Inside the research group, it has been developed several mobile platforms with different sensors and actuators as testbed, based on different technologies (Armesto et al. 2003). These experimental vehicles, as the one shown in Figure 1, are used for evaluating control, data fusion, obstacle detection and collision avoidance algorithms.



Figure 1: Real Vehicle.

In particular, control algorithms have been extensively validated on the vehicles (Armesto and Tornero 2003) and now we are including obstacle-avoidance algorithms to perform teleoperated navigation as well as autonomous navigation.

In this context, simulation has become a useful tool for evaluating the goodness of vehicles control systems and obstacle avoidance algorithms.

Two possible applications for these collision detection and obstacle avoidance techniques have been included: 1) simulation of an autonomous mobile robot moving along a prestablished trajectory and 2) a teleoperated mobile robot traveling around the environment and driven by the user. In both situations the autonomous/teleoperated vehicle will avoid moving obstacles in the environment by means of collision detection and obstacle avoidance algorithms.

The rest of the paper is organized as follows: in the next section, we describe the simulation environment in detail.

The following section explains the integration of collision detection and obstacle avoidance algorithms in the virtual environment. The application of an algorithm based on the computation of the Minimum Translational Distance is then presented. The paper ends with comments on possible applications and conclusions.

SIMULATION ENVIRONMENT

In this section we will describe the virtual environment created for testing collision detection and avoiding algorithms for mobile robots. We will also describe the possible configurations of the environment in order to simulate different situations.

Programming Software

A commercial programming software called DarkBasic® is used for modeling the environment. DarkBasic® is essentially a language that features many of the structures and commands found in the traditional BASIC language. DarkBasic® offers powerful specific commands to represent and move 3D objects and handle them with control devices such as mice, joysticks, steering wheels, etc.. It also provides an interface between DirectX commands and the programmer.

This software is basically oriented to the development of 3D games rather than professional applications. The main advantage of this programming language is that any kind of object is easily integrated. It allows the use of a wide range of graphic object formats, 3D (X, 3DS, MD2, MD3,...) and 2D formats (DDS, JPEG, BMP, TGA,...).

Another important feature is the possibility to expand its functionality using Dynamic Link Libraries (DLL) added to the programs.

Environment Description

The developed environment represents a typical four-lane highway where vehicles can travel in the same or opposite direction.

The ground surrounding the highway lanes has been generated randomly.

Five different vehicle models have been created for the simulation, using the popular 3D graphics tool 3D Studio MAX®. A motorbike represents the autonomous/teleoperated vehicle and four other car models are used for simulating the obstacles. The whole environment is displayed in Figure 2.

The number of obstacles is configurable. At the beginning of the simulation all the car models (moving obstacles) are loaded in memory as part of the configuration. The configuration also includes the order of appearance, the kind of vehicle, the initial lane, the speed and direction. These parameters are randomly generated using a random

number generator, a structural component of every simulator (Ingalls 2001).



Figures 2: Simulation Environment

For simulation time, other parameters have to be prestablished at the beginning of the application, as shown in Figure 3.

There are two running modes: 1) an algorithm is tested in different simulations or 2) different algorithms are compared for the same set of situations. This is obtained using different seeds or the same one when generating the random numbers.

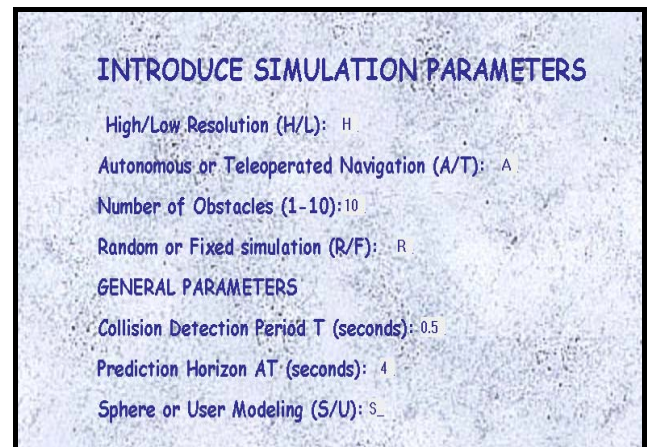


Figure 3: Configuration Parameters.

The autonomous vehicle follows a prestablished path, straight line by default.

All the obstacles involved in the simulation are created and stored in a buffer prior to the simulation begins. They are taken from the buffer as required, avoiding time consuming during simulation.

In the simulation, the obstacles start to appear, one by one, in front of the autonomous/teleoperated vehicle or at the rear part of it, at a certain distance.

There is a limit on the number of objects that can be represented by the system at the same time depending on

the computer hardware. The configured value for the number of obstacles is set to ten by default. It is possible to increase the number of moving obstacles by reducing the models resolution.

In order to save memory and reduce time consumption, the highway is considered as a circular buffer, where moving obstacles (other vehicles) are disappearing at the foreground/background and reappearing at the background/foreground respectively, keeping their movement parameters (speed and lane).

There are two possible camera views in the environment: back and lateral views. In the back view, the camera is located at the rear part of the autonomous/teleoperated vehicle. It focuses on the vehicle and the far background. In the lateral view, the camera points to the left side of the vehicle, this allows to see the objects behind the vehicle.

The present simulation has been executed on a desktop computer with the following features: Intel Pentium IV processor running at 1.8 GHz, 1 GB of RAM, 64 MB 3D graphics acceleration board ATI Radeon 8500.

INTEGRATION OF COLLISION-DETECTION AND OBSTACLE-AVOIDANCE ALGORITHMS

The simulation code has been designed to include collision detection and avoidance algorithms as independent modules.

The modular structure proposed is achieved by using Dynamic Link Libraries. As is well known, a DLL is a function collection implemented in a high-level language that can be loaded by any Windows application. This property gives a certain level of independence to the code stored in the DLL because it can be developed with any

high-level language and executed in a different high-level language environment.

This code independence is an interesting feature as it makes it possible to integrate the collision detection code in a simulation environment as well as in a real application. The algorithm implemented as DLL has to exchange data with the simulation environment and with the real application if required.

The diagram in Figure 4 shows the basic structure of the simulation program. It represents the case of an autonomous vehicle following a predefined trajectory (straight line).

In case of a teleoperated vehicle, the user establishes the references for movements but those orders will be overridden if collision is predicted.

Input-Output Parameters

The application creates the virtual environment and exchange data with the algorithm through some input and output parameters explained in the following paragraphs.

Input Parameters

Input parameters are those variables supplied by the environment to the algorithm. We can distinguish three kinds of parameters depending on their scope: general parameters, autonomous/teleoperated vehicle parameters and obstacles parameters.

1. **General parameters**, listed below, are in connection with the representation and the execution period (frequency) of the algorithm, introduced by the user, in configuration time (at the beginning of the simulation).

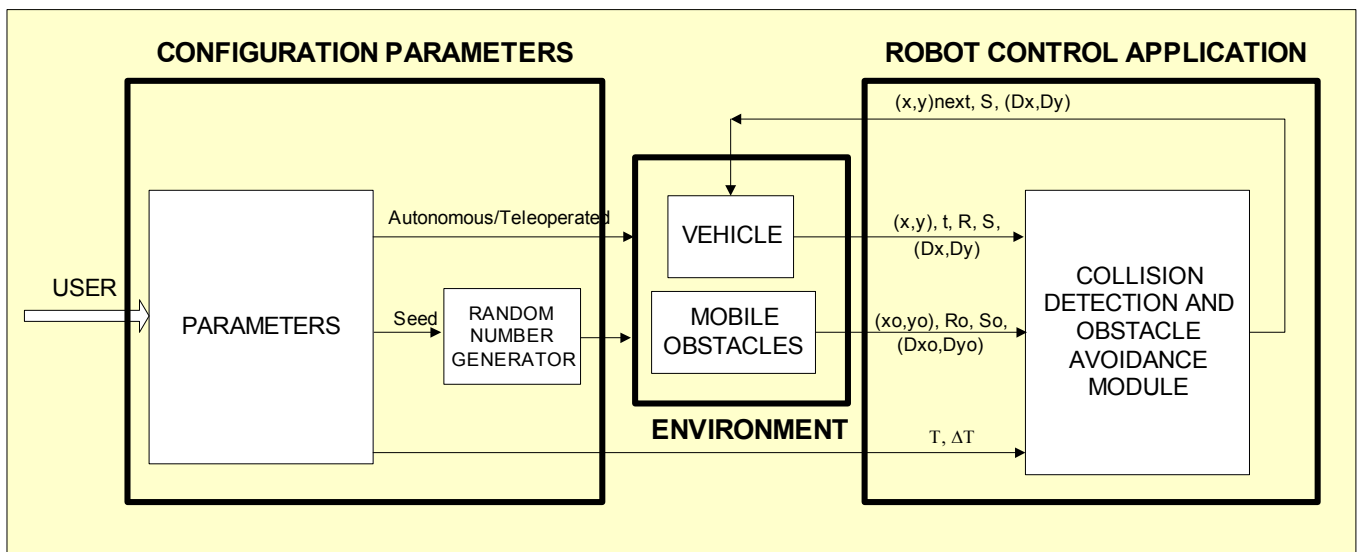


Figure 4: Simulation Program Structure

- 1.1. **Collision detection period T** (seconds) establishes the execution frequency of the collision detection and obstacle avoidance algorithm. It depends on the highest sampling frequency at which sensors can provide information from the environment. By increasing the period, it is possible to simulate the algorithm behavior with slow-dynamics sensors. However, if the period is too long, every obstacle may not be detected.
- 1.2. **Prediction Horizon ΔT** (seconds) is an essential parameter in order to predict the position of the autonomous vehicle and obstacles. Its default value is set to $\Delta T = 3 \cdot T$ but it can be modified by the user in the configuration time. It should be closely related to the objects speed.
2. **Autonomous-teleoperated vehicle parameters** refer to the vehicle model. By default, the collision detection algorithm uses a spherical model for the objects. It is possible to use other object models introducing user parameters at the beginning of the simulation.
 - 2.1. **Position (x,y)** (meters). They are the coordinates of the reference point of the vehicle. It may be the position of the center of gravity or any other characteristic point. When considering enveloping spheres, its center is usually considered as reference point.
 - 2.2. **Dimension parameters** (meters) are parameter that establish the width, length and height dimensions of the vehicle (w, l, h). If the vehicle is just as a sphere an only parameter (radius R) is enough to characterize the vehicle size.
 - 2.3. **Present instant of time t** (sec). It is taken the same for the autonomous vehicle and for the obstacles.
 - 2.4. **Vehicle speed S** (m/s) at present instant of time t .
 - 2.5. **Direction vector (D_x, D_y)** . Unit vector that indicates the direction to the target or the trajectory to follow.
3. **Obstacle parameters** refer to the obstacles displayed on the screen (detected by the sensors in a real application). They are very similar to the autonomous/teleoperated vehicle parameters.
 - 3.1. **Number of obstacles N** displayed on the screen or detected by sensors in a real environment.
 - 3.2. **(x,y) position** (meters) is similar to the autonomous vehicle position parameters for every obstacle.
 - 3.3. **Dimension parameters** (meters) are analogous to the vehicle dimension parameters for every obstacle.
 - 3.4. **Obstacle speed S_o** (m/s) in the present instant of time.

- 3.5. **Direction vector (D_x, D_y)** is a unit vector indicating the obstacle trajectory. It can be estimated from real data supplied by sensors or from simulation.

Parameters described above have to be collected each time collision detection and obstacle avoidance algorithms are called.

The simulation environment can be substituted by a real robotics application. Therefore, if the algorithm is going to be integrated on a real system, every input parameter will have to be supplied by a high-level planner. This means that the planner will have to take the information from sensors located on the vehicle.

Output Parameters

Output parameters are those variables returned from the algorithm to the virtual environment after each evaluation is completed. They are referred to the next instant of time ($t+T$) and state the best trajectory to avoid the obstacles. The three returned parameters are:

1. **Position (x_o, y_o)** (meters) corresponds to the coordinates at the next instant of time. The vehicle must be positioned there in ($t+T$) to avoid the nearest obstacle.
2. **Vehicle speed S_o** (m/s) in next instant of time ($t+T$).
3. **Direction vector (D_x, D_y)** is the unit vector indicating the direction to avoid the obstacles.

As these parameters are referred to a future instant of time and there is no information about the vehicle movement between instants t and $t+T$, an interpolation is usually needed. At the moment, linear interpolation has been developed but a higher-grade interpolation is also considered for the future.

APPLICATIONS AND CONCLUSIONS

The simulation environment has been used to test the behaviour of a collision detection and obstacle avoidance algorithm based on the computation of the Minimum Translational Distance [Bernabeu 2001].

This algorithm uses spheres for distance computation between vehicles as a way of reducing computational time. In particular, (Figure 5) shows how bi-spheres are considered for modeling the volume swept by each vehicle.

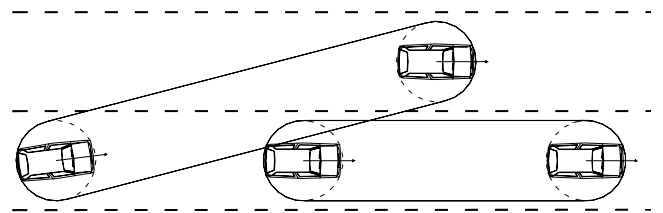


Figure 5: Bi-spheres Representing the Motions of Vehicles.

The algorithm provides the minimum translational distance of separation (positive value of MTD) or penetration (negative value of MTD) between two objects, as well as the direction where collision might occurs.

For each moving obstacle, an MTD is computed as a way of detecting possible collision. The obstacle with maximum negative MTD will be the one to be avoided first. That is, the algorithm carries out local collision avoidance.

In Figure 6, a set of pictures shows the sequence of movements performed by the autonomous and the other vehicles. The corresponding AVI file, showing the avoiding maneuver, can be downloaded from <http://ttt.upv.es/~marmoag>.

REFERENCES

Armesto, L. and J. Tornero. 2003. "Transport Process Automation with Industrial Forklifts." Accepted on 7th *International IFAC Symposium on Robot Control (SYROCO)*, Wroclaw, Poland.

Armesto, L.; J.C. Torres; J. Tornero. 2003. "Hardware Architectures for Mobile Robots" Accepted on 7th *International IFAC Symposium on Robot Control (SYROCO)*, Wroclaw, Poland.

Barton, R. R. 2001. "Designing Simulation Experiments". *Proceedings of the 33rd Conference on Winter Simulation*, pp. 47-52, Arlington, Virginia, United States.

Bernabeu, E.J.; J. Tornero; M. Tomizuka. 2001. "Collision Prediction and Avoidance Amidst Moving Objects for Trajectory Planning Applications". *IEEE Press*, pp. 3801-3806. I.S.B.N.: 0-7803-6475-9/01.

DarkBasic Software Limited 2000. "DarkBasic Reference Manual."

Ingalls, R.G. 2001. "Introduction to simulation". *Proceedings of the 33rd Conference on Winter Simulation*, pp. 7-16, Arlington, Virginia, United States.



Figure 6: Avoiding Maneuver Implemented on the Simulation Environment.

SIMULATION IN LOGISTICS

Logistics Simulation

A PARALLEL PROCEDURE FOR A DISTRIBUTED LOGISTICS SCHEDULING

Wilhelm Dangelmaier
University of Paderborn
Fürstenallee 11
33102 Paderborn, Germany
E-mail: whd@hni.upb.de

Hubertus Franke
University of Paderborn
Fürstenallee 11
33102 Paderborn, Germany
E-mail: franke@hni.upb.de

Peter Scheideler
University of Paderborn
Fürstenallee 11
33102 Paderborn, Germany
E-mail: peter.scheideler@hni.upb.de

KEYWORDS

Logistics cooperation, parallel IT-infrastructure, PC-cluster, optimisation, cost functions.

ABSTRACT

To increase the effectiveness of transportation, logistics companies could cooperate by joining all their transportation orders, sharing their resources (trucks etc.) and integrating their IT-Infrastructure. This infrastructure could be shared by building up PC-Clusters that can solve problems as parallel computer-systems. Within this systems the transportation orders will be matched with all relevant resources. In this process it cannot be avoided that a company may lose an order to another company with free capacities that could handle the order more effectively. To still generate a fair procedure where both parties can win, this paper analyses approaches where companies will pay provisions to those companies that have lost orders within this optimisation process. Furthermore we have implemented a possible scenario of a cooperative logistics system by using PC-Clusters and analysed the results.

INTRODUCTION

Due to increasing competition more and more logistics companies have to co-operate with each other to use the synergy effects of a global market. It is therefore a sensible action of the cooperating companies to pool their computer networks to a total net in order to solve problems which require intensive distributed computing, e.g. in the field of transportation scheduling. The special research field SFB376 (SFB376-ol), funded by the German Research Community (DFG), examines how distributed systems in the context of a real time production and logistics control can be effectively used. Thereby strategies are to be examined that propagate a gradual change between "generation of a valid solution" and "improvement of this solution" in the field of a real time transportation planning and control system. The computation of a valid plan aims primarily at a fast adjustment to a changed situation, e.g. in case of a disruption. Here it is crucial that on the basis of the given general conditions a valid solution must be quickly computed in order to initiate changes in the ongoing process. If there are no changes, there is still the possibility of a continuous improvement of the existing plan.

With the continuous improvements a global optimisation should be reached in which the principal objective is to find a global improvement of the existing solution for all involved co-operation partners. Here it is generally possible that orders are moved among the logistics companies by means of improvement routines, which can result in advantages or disadvantages for the individual company. Especially in the case of disadvantages strategies are to be developed, which offer companies an incentive for taking part in such co-operation networks by granting them compensations.

THE PROBLEM OF CONTROLLING DISTRIBUTED LOGISTICS NETWORKS

If logistics service companies cooperate in order to organize transfer orders more efficiently and cost-effectively, processes are to be developed which include the following objectives:

1. The cooperation within a logistics group must guarantee a (global) plan improvement with regard to cost-savings and to a better availability of resources.
2. Participating in the network may not punish the individual partners.

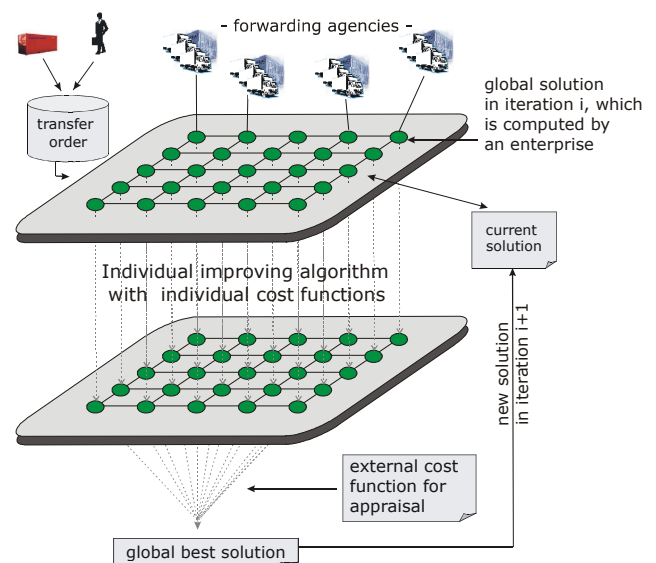


Figure 1: Scheduling within cooperating networks

An improvement of the transportation scheduling can only be effectively achieved if all involved partners supply their information about available resources and orders so that a transportation scheduling can be accomplished by means of centralized procedures. Therefore algorithms and procedure models are to be developed, which achieve a significant improvement in the transport scheduling under the consideration of the above-mentioned objectives.

STATE OF THE ART

In the very extensive literature of the controlling of distributed production and logistics networks (Busch et al. 2002) there is no approach known which proposes the use of distributed systems, both for controlling of sub-networks and for the controlling of the total net. This is quite understandable since sub-problems of distributed production and transportation logistics are already NP-complete (Oertel 2001) and therefore sequential problem solutions are overextended with the guarantee of a real time capability (Zäpfel and Wasner 2002). Thus problems in the controlling of distributed logistics systems are hardly to run parallel. However, there is much unused computation power within a company in form of PCs that do not operate at full capacity. By using these additional resources a speedup (Censor and Zenois 1997, p.23), i.e. the absolute gain of computing time by employing distributed systems, could be achieved. In (Florian 1993; p.23 ff.) researches on supply chain management provide a solid basis in particular for merely business aspects of this paper.

Fischer et al. (1993) describe the possibility of enormous savings both for business and economic assets by modifying the actual order processing by forwarding agencies with the objective of an integrated supply chain management. A study on the improvement of the efficiency in the transportation sector aims therefore at the utilization of obviously unused transportation capacities. Supply chain management is currently considered as one of the most effective instruments for the optimisation of distributed logistics processes. It includes all methods and resources for the organization and co-ordination of logistics networks by the organization of the value-added network, flow orientation of individual companies, integration of the information basis and the collaboration in planning and control processes. (Kühner 2002) (Baumgarten 2002) (Bretzke 2002) The philosophy of the SCM is based on a continuous and efficient integration of all goods, information and flow of values from the first supplier to the final customer. Customer integration is thereby as important as the integration of the intra-corporate functions and the integration of suppliers. (Wagner 2002) Against this background SCM aims at bridging the interfaces that emerge due to labour division and simultaneously benefits from the expertise of external partners by their frictionless coordination and/or integration into internal processes. The individual companies within a network are thereby to be coordinated under cost restrictions in such a way that product and waiting queues become as small as possible and the respective customer is still optimally supplied. (Bodendorf et al 2001) (Straube 2002) (Weber 2002) (Zäpfel and Wasner 2002) In order to solve these distrib-

uted problems, which are additionally subject to many restrictions, partly unrealistic simplifications have to be made in order to use central algorithms. A possible solution is the use of partial-intelligent agents (PIAs). PIAs are defined as relatively small computer units that allow employing standard models for well-defined fields of functions as modules of simple algorithms. The agents negotiate with one another, whereby they can cooperate or compete (market mechanism) (Falk et al. 1993)(Kirsch 2002). During the last years the requirements have risen to such an extent that the existing Enterprise Resource Planning (ERP) and production planning and controlling systems (PPS) couldn't meet these anymore, which results particularly from outdated modes of operation and their internal focus. Advanced planning systems (APS) are to remedy these deficits, which offer, as modular structured information systems, extensive transactional and decision support for distributed planning. Modules of APS support the configuration of supply chains by modelling and simulation tools, tactical-operational planning with demand, production and transportation functionalities as well as the execution of transactions and monitoring throughout the whole supply chain (Krüger and Steven 2002). However, if there is no centralized planned and controlled supply chain, but only equal enterprises and the supply chain which is organized in autonomous organization units, then only a collaborative, decentralized and self-coordinating planning of success potentials is target-oriented. This collaborative form (CSCM - Collaborative Supply Chain Management) of the co-operation of supply chain enterprises includes in particular the open information exchange of data between customers and suppliers, which is based on trust, and which results in a synchronized planning and a distributed coordinated control in case of a breakdown (Busch and Dangelmaier 2002) (Bretzke 2002) (Mors 2002). (Truszkiewicz 2002) suggest a concept that proposes the organization of logistics services by internal logistics operations, which were originally handled by external service providers.

A CONCEPT FOR THE CONTROLLING OF HETEROGENEOUS NETWORKS

Decentralized and centralized improvement procedures formed the algorithmic basis for the development of a distributed, real time improvement procedure for cooperative distributed transportation networks, in which the partners align with others in order to plan global distributed transport inquiries without offending the interests of the others. Each partner can plan a received transport inquiry according to his personal or the overall interest. A global control function then chooses the best available solution.

Heuristics for distributed transportation disposition

The heuristics in the distributed improvement procedure are now closely examined: Figure 2 shows the structure of the applied heuristics, which can be divided into four groups: The first heuristic element indicates two procedures that initiate improvements along the route of one transport mean.

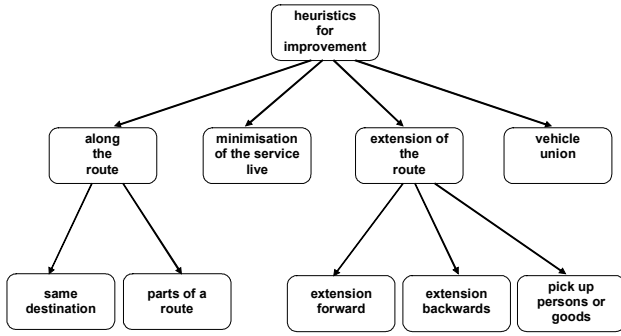


Figure 2: Applied heuristics for the transportation disposition

On the one hand all transportation objects (persons, goods), which simultaneously are at the same station and (with shared time windows) aim for the same destination, should be transported together in one transport mean. On the other hand this procedure, in consideration of customer restrictions and time windows, is also to be applied for all sub-routes.

A further possibility exists in the extension of current routes. In this heuristic process the routes of certain transport means are modified in consideration of given time windows and the utilization of existing buffers. Thereby three proceedings can be differentiated. A transport mean can extend its route in order to satisfy an additional customer request. Both, forward extension and backward extension, are tolerated. Furthermore, a detour to a neighbour station is possible. However, the necessary buffer time must be sufficient and the given restrictions must be observed. The formation of vehicle unions (e.g. truck with trailer) is a separate sub-heuristic. If it can be arranged that one transport mean can drive together with another transport mean in a convoy, then this is done in order to save energy and to reduce costs. The function of the minimization of buffer time at the stations appears implicitly in the described heuristics. If there are buffer times at the stations during the improvement, then this function is to make an appropriate allocation, which minimizes the buffer times. There is the possibility to book certain orders for other vehicles in consideration of the given time windows or to suggest an alternative departure time for these orders. Thus, the stand-by time of a vehicle and thus the buffer time in the system can be minimized.

Calculations of costs in a network

Let

$$C_E = C_{SD,E} + C_{Trd,E} + C_{SD,E}$$

be the total costs of a specific logistics company E, whereby $C_{SD,E}$ are the station and reloading costs, $C_{Trd,E}$ the costs depending on the route and $C_{SD,E}$ the vehicle operating costs.

Let $C(E_i)$ also be the total costs of all companies with

$$C(E_i) = C_{SD} + C_{Trd} + C_{SD}(E_i).$$

Then the main objective is to minimize the total costs $C(E_i)$ of all companies E_i . Each company E computes - in consideration of its individual cost functions - the total costs $C(E_i)$. Thereby all companies co-operate as a parallel cluster. The solution

$$C(E_i) = \text{Min} \{C_{E(i)} \mid C_{E(i)} \text{ are the total costs, which were computed by company } E(i) \forall i \in \{1, \dots, |E|\}\}$$

is realized afterwards and communicated to all companies for further improvement routines.

Guaranteeing that no company has disadvantages by the global improvement routines

Let

$$R = (st_1, tr_1, st_2, tr_2, \dots, st_{n-1}, tr_{n-1}, st_n)$$

be a route from station st_1 to station st_n via the route tr_1 to tr_n , which serve as a basis for the pricing of a transfer order.

Let $I_{E,0}$ be the total revenues of the company E for the regarded planning horizon at the underlying initial solution. Let $I_{E,i}$ be the computed total revenue of a company E after i iterations. Let $C_{E,0}$ be the total costs of a company E of the initial solution. Let

$$G_{E,0} = I_{E,0} - C_{E,0}$$

be the first computed profit of a company E. Then it is to be guaranteed that the new solution distributes the transport volume fairly in all improvement steps to all transportation networks involved (e.g. each individual company is not worse off than in its initial solution): Then the following applies to all regarded transport companies E:

$$\text{If } G_{E,0} > I_{E,i} - C_{E,i} \text{ for any } i \in \mathbb{N} \text{ with } i > 0,$$

then E must get a monetary compensation from the company that obtained a profit by the improvement.

This could take place in form of a premium paid by those companies that benefit from the total optimisation. Let $P(I_{A,0})$ be the commission that company A must pay to company B that was disadvantaged by the optimisation, then the profit allocation after i+1 iterations is:

$$\begin{aligned} G_{A,i+1} &= G_{A,i} + I_{B,i} + C_{B,i} - P(I_{A,i}) \\ G_{B,i+1} &= G_{B,i} - I_{B,i} - C_{B,i} + P(I_{A,i}) \end{aligned}$$

Here it is assumed that all companies incur the same operating costs.

IMPLEMENTATIONS

As already mentioned above each enterprise involved in the network is described as an autonomous process in form of a software agent. Physically seen, it is therefore possible that each enterprise is represented by a computer in the distributed PC cluster. Through the interaction of these agents a first solution is computed with simple algorithms.

Because results, which are achieved during an agent-based problem-solving process, are not rejected, we speak of a greedy solution. Though a generated greedy solution meets

the previously specified requirements, it generally offers further improvement potential. In the context of a real-time controlling of the logistics network a change was realized from the greedy solution to the improvement procedure. For the gradual improvement steps the agents form clusters, which improve the already existing solution by a local-search-procedure. Within our sub-project C2 of the SFB 376 we centred our research around the procedures „Simulated Annealing” (Feldmann and Kistner 2000, p. 489), genetic algorithms (Holmqvist et al. 1997, p. 283), and tabu-search- algorithms (Reeves 1996, p. 14) for distributed improvement procedures for the transportation scheduling. Already after the first prototypical implementations the best results were obtained with tabu-search. Since the implementation of a tabu-search procedure is generally less complex than the implementation of genetic algorithms or simulated annealing, the focus for the further investigations was placed on tabu-search. The employment of tabu-search is not limited to a certain problem area, but can be applied with small modifications to different optimisation problems. For the cycles of the program an initial solution with 1.828 transportation objects (cargoes) and also 1.828 vehicles was generated. In the next step the initial solution was tested in a parallel cycle with different tabu depths and a fixed number of processors. The choice of the tabu depth depends on the size of the problem and would have to be adjusted to problems of other sizes.

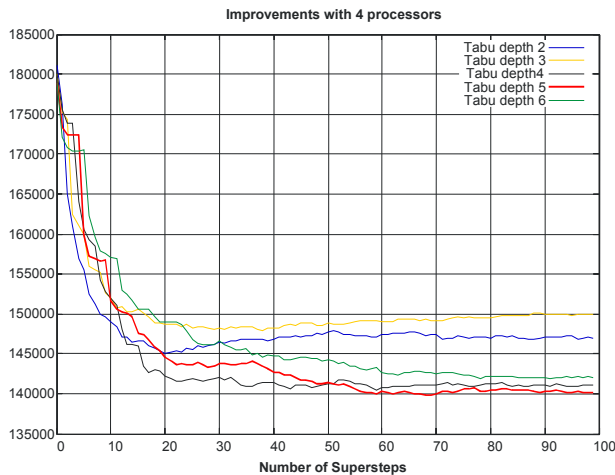


Figure 3: Costs with variable tabu depth

It turned out that the tabu depth 5 offers the best results for this problem. After the determination of this tabu depth for this problem size, several tests with a different number of processors were made. The number of supersteps amounted to 100 in each cycle with a duration of five seconds for each superstep. The results are shown in Figure 4.

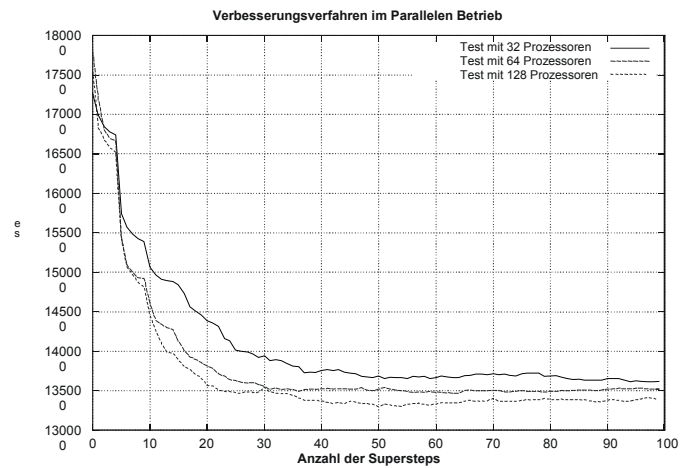


Figure 4: Tests with 32, 64, 128 and 16 processors and tabu depth 05

After looking at the costs reduced by the optimisation with a variable number of processors it becomes clear that improved results were achieved by increasing the number of processors involved in the optimisation process and the supersteps. Moreover, after a certain number of supersteps no substantial improvements can be obtained, partly even worse results can be noticed. Figure 5 shows the shift of the costs with each doubling of the number of processors and the tabu depths three and five and each 100 supersteps.

If one compares the improvement of the cost units that were produced by applying only two processors (with the tabu depth 05) with the result delivered from 128 processors, then a difference of 8.242 cost units can be noticed. This corresponds to an improvement of 5,531%.

#Processor	128	64	32
tabu depth 03	-	138815	137855
Tabu depth 05	132145	135192	134644

#Processor	16	08	04	02
tabu depth 03	139986	143260	138486	149007
Tabu depth 05	137487	137815	140182	140387

Figure 5: Comparison of the results with the tabu depths 03 and 05

By the application of the chosen tabu-search-procedure as a superior meta-heuristic in the improvement software with several smaller heuristics, which were conceived particularly for a distributed transport system, it could be shown that by paralysing the system clearly better results could be obtained. While the improvement procedures could only partly supply good results in a sequential case and/or in cases with very few processors, the probability of a good solution increases significantly with the employment of several processors. This fact can thus point to the immense solution space, which in fact does not reduce during the improvement procedure; but the number of relevant

neighbourhoods, which promise a further improvement, decreases rapidly.

If one starts a procedure sequentially, then in most cases at a certain point there will be no further neighbourhoods, with which costs can be lowered. By using several processors, which have sufficient time for individual supersteps, several neighbourhoods, which lie on the "solution paths" can be visited and examined. By the fact that the call parameters of the individual sub-heuristics are randomly selected, individual processors automatically scan the same neighbourhood and generate the same results after a superstep. In order to avoid this situation, either the strategy of the selection procedure can be changed or the number of processors is reduced. Here it should be considered that a deterioration of the situation is possible. Hence a possible "overlapping" of scanned neighbourhoods in this system cannot be excluded. Which number of processors is most appropriate, depending on the problem size and solution quality, must be examined for the individual problem.

CONCLUSION

In this scenario different logistics companies formed a network, in order to cooperatively organize transportation scheduling. The processors in the considered scenario represented the PCs in the individual company net. With distributed improvement procedures all problem-relevant information was supplied to each processor and each processor organized improvement algorithms for a given time period according to individual and global procedures. In a synchronous distributed model the computers exchanged their results according to fixed time intervals. The most cost-effective solution was communicated to all computers as the new actual plan. In order to get, if possible, different heuristics for individual processors for the employed tabu-search-procedure, each processor can use individual heuristics to achieve the desired objectives. The scenario was simulated under laboratory conditions by using the same heuristics for all processors, however the processors evaluated their own transport means with more favourable costs. If there are disadvantages for certain logistics service providers in the process of a global improvement, then these companies have to get a commission from those logistics service providers which benefited from the global improvement. The procedure was accompanied by extensive result analysis, so that, with an employment of 128 processors, after 100 supersteps a global improvement of the total costs of over 5 % was achieved compared to a "two processor procedure". Therefore both, efficient algorithms and software prototypes, were developed, which can be used for real-time hierarchical planning and controlling of logistics network systems.

REFERENCES

- Baumgarten, H.: Statement – Logistics Service Provider: Innovatoren in der Supply Chain? In: Supply Chain Management 3/2002 S.69ff
- Bodendorf, F.; Butscher, R.; Zimmermann, R.: Konzept zur agentengestützten Auftragsüberwachung in Supply-Chains. In: Logistik Management 2001 Ausgabe 2/3 S.13ff
- Bretzke, W.: Statement – Implementierung von SCM-Informationssystemen. In: Supply Chain Management S.61ff
- Bretzke, W.: SCEM – Entwicklungsperspektive für Logistikdienstleister. In: Supply Chain Management 3/2002 S.27ff
- Busch, A.; Dangelmaier, W.: Collaborative Supply Chain Management – Grundlagen, Konzepte, Systeme. In: Supply Chain Management 2/2002 S.15ff
- Busch, A.; Lange, H.; Langemann, T.: Marktstudie: Standardsoftware zum Collaborative Supply Chain Management, Dangelmaier W. (Hrsg.), ALB/HNI-Verlagsschriftenreihe, Paderborn 2002
- Censor, Y.; Zenois, S. A.: Parallel Optimization: Theory, Algorithms and Applications. Oxford University Press, New York, Oxford, 1997
- Falk, J.; Spieck, S.; Mertens, P.: Unterstützung der Lager- und Transportlogistik durch teilentintelligente Agenten, in: Informationsmanagement Ausgabe 2/93
- Feldmann, M.; Kistner K.-P.: Heuristisches Problemlösen in Analogie zur Natur. In: WiSt (2000) 9, S. 489-493.
- Fischer, K.; Kuhn, N.; Müller, H.-J.; Müller, J.P.; Pischel, M.; Schroth, A.: Verteiltes Problemlösen im Transportwesen. In: Informationsmanagement 2/93 S.32ff
- Florian M.: Fahrerlose Transportsysteme – Chancen und Risiken der Automatisierung des innerbetrieblichen Transports. IUK-Institut für sozialwissenschaftliche Technikforschung, Dortmund, 1993
- Holmqvist, K.; Migdalas, A.; Pardalos, P. M.: Parallelized heuristics for combinatorial search. In : Migdalas, A. u.a. (Hrsg.): Parallel Computing in Optimization. Kluwer, Bosten, Dordrecht, London, 1997, S. 269-294.
- Kirsch, C.: Agent Networks with a License for Company Success. In: Supply Chain Management 2/2002 S.29ff
- Krüger, R.; Steven, M.: Advanced Planning Systems – Eine neue Generation von SCM- Informationssystemen. In: Supply Chain Management 2/2002 S.7ff
- Kühner, M.: Fourth Party Logistics Service Provider. In: Supply Chain Management 3/2002 S.21ff
- Mors, M.: Supply Chain Event Management Systeme. In: Supply Chain Management 2/2002 S.25ff
- Oertel, P.: Routing with Reloads, Dissertation, Mathematisch-Naturwissenschaftliche Fakultät der Universität zu Köln, Köln 2001, S. 14 ff
- Reeves, C. R.: Modern Heuristic Techniques. In : Rayward-Smith, V. J.; Osman, I. H.; Reeves, C. R.; Smith, G. D. (Hrsg.): Modern Heuristic Search Methods. John Wiley and Sons, Chichester, New York, Brisbane, Toronto, Singapore, 1996, S. 1-25.
- SFB376 online: <http://www.upb.de/sfb376/projects/c2/>
- Straube, F.: E-Logistik – Grundlagen und strategische Perspektiven. In: Supply Chain Management 3/2002 S.7ff
- Truszkiewitz, G.: Insourcing logistischer Dienstleistungen als Supply Chain Strategie. In: Supply Chain Management 3/2002 S.55ff
- Wagner, S.: Gestaltung von Beziehungsmustern in Supply Chains. In: Logistik Management 4/2002 S71ff
- Weber, H.: Asset Optimization im Fokus der strategischen Netzwerkoptimierung. In: Supply Chain Management 2/2002 S.39ff
- Zäpfel, G.; Wasner, M.: Strategische Gestaltung internationaler Logistiknetzwerke für den flächendeckenden Stückgutverkehr durch Kooperation. In: Logistik Management 4/2002 S.53ff
- Zäpfel, G.; Wasner, M.: Strategische Gestaltung internationaler Logistiknetzwerke für den flächendeckenden Stückgutverkehr durch Kooperationen, Logistik-Management, Ausg. 4, 2002

SIMULATION OF A CONTAINER TRANSPORT SYSTEM BETWEEN PORT AND INLAND TERMINAL DEPOTS

Leire Maruri
Laura Núñez
Ana Vidarte
José M. Ezquerro
Ana Martínez
IKERLAN Research Centre
E-20500 Arrasate-Mondragon, Spain
E-mail: lmaruri@ikerlan.es

KEYWORDS

Container Terminal, Inter-modal transport, Traffic Control, Line Management System, Host Computer, Shuttle, Real-time simulation, Rail, Swap Area.

ABSTRACT

This document describes the development of a simulation of an automated container transport system that connects a harbour terminal and an inland depot with dedicated railways. The simulation has been developed as part of the European project ASAPPOne (Intelligent Shuttle Fleet Connecting Split Sea And Inland Container Storage Areas), of which IKERLAN is a partner.

The aim of the ASAPPOne project is the automation of container transport tasks between container terminals (front/end areas). It consists of a fleet of shuttles, which run in convoy mode on train rail tracks along the connecting path and autonomously on a flat surface of coated concrete in the front/end areas.

A Traffic Control System (TCS) receives container transport commands from a Host Computer and calculates the route for each shuttle inside the terminal. Along the railway, a Line Management System (LMS) calculates the route for each convoy of shuttles. In order to test and refine the Traffic Controller and the Line Management System, a simulation of all the elements that interact with it (cranes, shuttles, crossing controllers, swap areas, Host Computer) has been designed and developed.

INTRODUCTION

Most European seaports are surrounded by urban spaces or down town, what makes almost impossible to embark on significant expansion projects. Because of the impossible growth of these second rank port areas, the small/medium size container terminals located in these kind of ports cannot cope with the traffic volume (Gambardella et al., 1998) required to guarantee a sustainable exploitation of the equipment required to attract the shipping lines (at least 4 berth cranes).

At the same time, a high performance berth system able to grant the economical figures to run a container terminal if

operating in an urban environment, will create huge traffic and pollution problems (Fantuzzi-Reggiane et al., 2000).

The volume of economically sustainable traffic for a Container Terminal in the near future is completely in conflict with the road network. The rail tracks connecting the port, because it has to deal with various users, cannot be enough exploited for containers. Moreover the road network cannot be modified accordingly to the heavy trucks circulation requirements.

In conclusion, to guarantee the port efficiency (its survival), it will be essential to solve the problem of the container flow from the port area to the inland, to avoid to lose the advantages achieved in the berth performances. Another question closely related to high productivity is the yard space required in the terminal to stack the boxes taking into account the dwell time.

The realistic solution is represented by an inland depot (dry port) located in a suitable peripheral site, well served by regional and international railway, road and/or inland waterways networks (just as the inter-modal continental nodes). This inland depot must now be connected in real time to the port activity reducing the port yard stack just as a daily buffer.

The two areas of this new SPLIT terminal could be easily linked by a dedicated single rail track (new or existing). However, today's railways are not efficient enough to meet the requirements of a high performance system, whereas an automated transfer is the right answer to the question (direct ship to train transfer, in the same day). This solution is strategic for the sustainable growth of the container traffic from Europe to overseas (Fantuzzi-Reggiane et al., 2000).

The ASAPPOne objective is to develop a new "transfer wagon" based on some solutions achieved in a previous European project (ASAPP: Automated Shuttle for Augmented Port Performance), but having innovative and enhanced performances, especially devoted to a new split terminal (see Figure 1) and able to maximise the path use (the distance being 5 to 50 Km) just in the way to make the connecting path the "compensation buffer" between the two parts of the terminal while their on time productivity is too different.

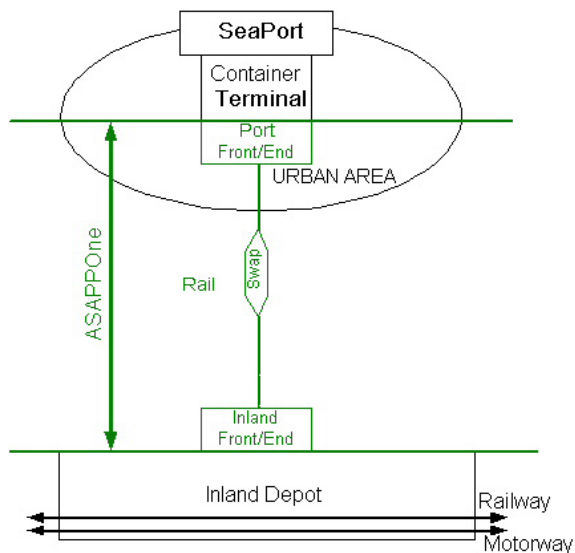


Figure 1: ASAPPOne System

As well as exploiting in the sea transport chain, it is studied the possibility to adapt the ASAPPOne System to other multimodal links, like the transportation of containerised waste from fluvial port to treatment workshop.

Within the framework of the Growth European research programme, IKERLAN takes part in the ASAPPOne project along with the following companies: FANTUZZI-REGGIANE, NOELL Crane Systems, SCIRO, CYBERNETIX, SUDITALIA Terminal Operator, SITA and SALERNO Container Terminal. Inside the project IKERLAN is in charge of the development of the Traffic Control System and Line Management System and the implementation of the real time simulation.

Firstly, the paper will focus on a brief description of the physical elements that constitute the system layout. Secondly the paper analyses the control structure of ASAPPOne. Then, each of the simulated models in the simulator unit will be described. The paper continues with a brief explanation of the simulation execution and finally the conclusions from this work are summarised.

ASAPPOne SYSTEM DESCRIPITON

The first task to be dealt with is the definition of the functional requirements of the system. To achieve this first objective, it is mandatory to identify an initial scenario (Figure 2) to develop the base architecture. On the basis of this, it will be possible to adapt and customise the system to match all the different exploitation possibilities.

The System can be split into three main physical parts:

Port Front/End Area

The Terminal size may be envisaged to have a capability around 200-500,000 TEUs/years, i.e. a 2nd rank port equipped with 3 to 5 Ship To Shore cranes. The port front/end area, inside the ASAPPOne concept, will only be used to buffer functions. It is expected to have the berth

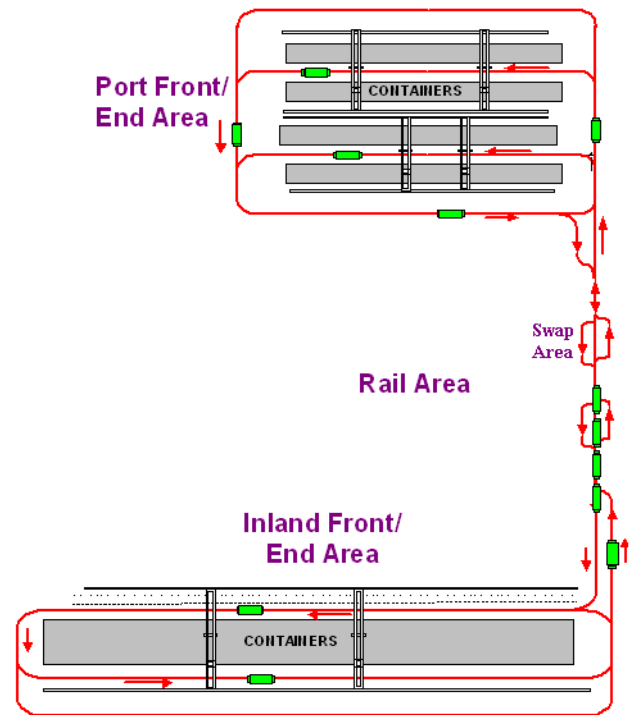


Figure 2: ASAPPOne system scenario

working for approximately the 40-50% of the time at high throughput, when the optimised exploitation of the connection or rail line implies a rather constant flow.

Inland Front/End Area

While the minimisation of the seaside buffer area is an object to be achieved, for its own nature this problem does not apply to the inland depot, where space is not to be considered a strong constraint.

Connecting Rail Area and Shuttle

The Connection Link (Rail Area), intended as sum of physical transfer media – shuttle – and functional management, is the actual scope of the research. To be able to cope with different geographic layout, the best way is to organise it on a modular base. The modularity is given by the swap areas and the distance between them. This swap areas allow to have convoys running in different directions within the railway. The first attempt layout configuration is a 8 Km long rail line with two intermediate 500 m swap areas. For what concern the automatic guided vehicle (the ASAPPOne Shuttle), the base arrangement will be the ASAPP shuttle, with some modifications to cope with new tasks. The vehicle will have to be able to transport up to 3 TEUs and run on railway tracks in addition to solid rubber.

As well as the physical parts, the control structure of the system must be defined.

CONTROL STRUCTURE

The control structure of the ASAPPOne system is shown in Figure 3.

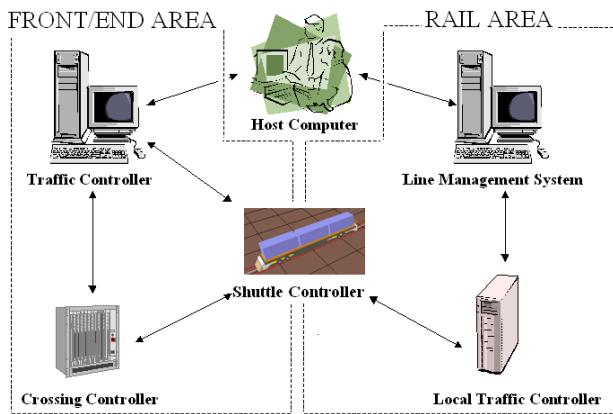


Figure 3: ASAPPOne control structure

As can be seen in Figure 3, the ASAPPOne control structure is first divided into two main subsystem controls (Front/End Area Control and Rail Area Control), which interact with an external control system (Host Computer). Each of the subsystems is in turn divided into three sub-sections: Traffic Control, Crossing Control and Shuttle Control in the Front/End Area (Ezquerria et al. 2000) and Line Management System, Local Traffic Control and Shuttle Control in the Rail Area. It should be taken into account that each front/end area (port and inland depot) will have its own Traffic and Crossing Controller.

The Host Computer (HC) manages general Storage Area activity. It controls the cranes and the position of the containers in the Port and Inland Storage Area. It assigns loading and unloading tasks to the Traffic Controllers in order to transfer containers from one front/end area to another (Ezquerria et al. 2000). It communicates with the cranes, sending them container load/unload commands and receiving information from them.

The Traffic Control System (TCS) is in constant communication with crossing and shuttle controllers of the front/end area that it controls. When it receives a transport request from the Host Computer, if necessary, it checks which shuttles are available in the area and selects the most appropriate (Ezquerria et al. 2000). It assigns them tasks and plans their routes according to the request and scenario (Evers and Koppers 1996). In this way, loaded or empty shuttles will be sent to an exit line of the front/end area to form a convoy of shuttles. Once a convoy is formed, it will enter the rail area controlled now by the Line Management system. But the TCS will also have to take charge of the shuttles that come into the terminal from the connecting rail line, and it will also assign them tasks in order to unload or load them.

Each crossing in the front/end areas has its own Crossing Control System (CCS), which regulates local shuttle traffic (Ezquerria et al. 2000). The TCS sends each local control system the list of tasks associated to the shuttles due to arrive at the crossing. When a shuttle arrives, it contacts the crossing controller and identifies itself. If the shuttle does not receive permission to continue on, it stops and waits until it does.

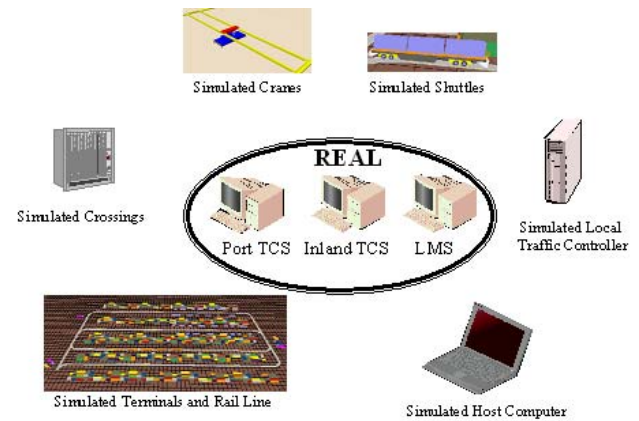


Figure 4: Traffic Controllers, Line Managements System and simulated elements

The Line Management System (LMS) is in charge of controlling the convoy of shuttles in the rail area. Being in communication with the Local Traffic Controller, it receives convoys position information. Whenever a convoy is prepared to exit from a swap area line (unidirectional), it asks for permission to enter the rail line (bi-directional), and for a route to follow. The LMS will at this moment check the situation of the rest of the convoys and assign the permission (Stop or Go On) and the best route (next swap area line identification) in order to avoid frontal collisions in the rail line. It means that the LMS will transfer the containers from an exit line of one container terminal to the entrance line of another.

The Local Traffic Control System (LTCS) physically monitors the traffic conditions on the rail area, communicates with the shuttle and switches and determines the local traffic control procedures to allow shuttles running nearby the swap area switches. The local traffic control system will be equipped with suitable sensors to determine the shuttle positions along the rail and to communicate with each of them. It acts as an intermediary between the LMS and the shuttles.

Finally, the Shuttle Control System (SCS) handles the basic instructions that it receives from the TCS, the CCS (Ezquerria et al. 1995) and the LTCS. These instructions consist of movement commands and load/unload commands. On one hand, the shuttles circulate autonomously in the front/end areas, but they have to avoid collisions with other shuttles, so they must be able to run in convoy mode. On the other hand, the shuttles run in virtual convoy mode on the rail area (constant distance between many shuttles running at the same speed and varying the speed accordingly to the remaining path, without any mechanical link). To operate in convoy mode, the shuttle is equipped with a sensor that is able to detect other shuttles in front of it.

SIMULATOR UNIT

For a demonstration of the ASAPPOne system, given the impossibility of using an entire fleet of real shuttles (Liu et al., 2000), two available container terminals and a complete railway connecting both terminals, a shuttle fleet, crossing controllers, cranes, local traffic controllers, host computer

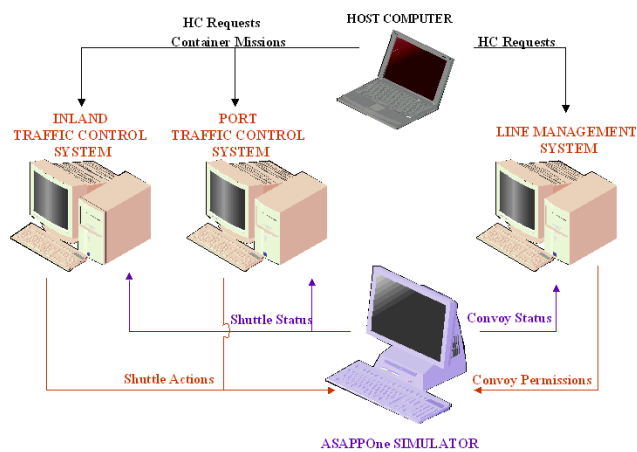


Figure 5: Distribution of controls

and terminal behaviour simulator (Figure 4) has been developed. This simulator, in conjunction with two real Traffic Controllers and a Line Management System, has been used to test and refine the whole system. Each Traffic Controller runs on one PC, the Line Management System on another and the simulation of the host computer on another. The simulation of the remaining elements, as well as a visual representation of their performance, runs on a fourth PC. The whole ensemble is configured as a HIL (Hardware In the Loop) system (Ezquerro et al. 2000) (Figure 5).

The tool used for the development of the simulation is QUEST, a flexible object-based discrete event simulation tool to efficiently model, experiment and analyse facility layout and process flow. 3D physical models are quickly created through pushbutton interfaces, dialog boxes and extensive libraries. Real-time interaction enables modification of model variables and viewing parameters during runs (QUEST 2001).

Each element in the simulation has its own set pattern of behaviour. Next, each simulated model behaviour is described.

Shuttle model

The shuttle has 3 TEUs carrying capability, and is able to handle ISO 20 ft, 40 ft and 45 ft size containers.

The shuttles models have been programmed to enable them to move along a straight line at a maximum speed of 50 Km/h on rails and 18 Km/h on concrete with an acceleration/deceleration of 0.15 m/s^2 . The speed at curves is reduced to 5 Km/h in concrete and to 10 Km/h on rails.

Each shuttle is fitted with a sensor that enables it to detect the presence of other shuttles on the path it is travelling on. In response to the information received through the sensor, the shuttle adjusts its speed to that of the shuttle in front, effectively forming a convoy.

Finally, the shuttle model implements all the real communications with the rest of the elements (simulated and real ones).

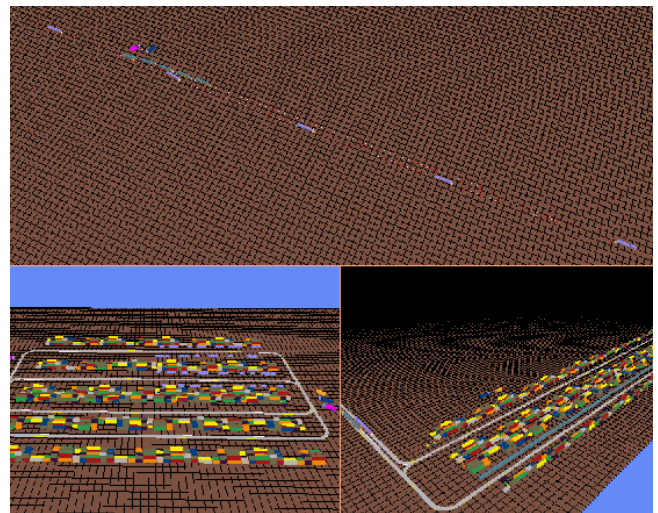


Figure 6: General view of an ASAPOne model

Crossing model

The crossings are modelled to control local traffic in the front/end areas, thereby avoiding collisions, and to transmit new commands to the shuttles. They detect when shuttles arrive to the crossings and work as traffic lights.

Cranes model

The crane models can manoeuvre themselves in line with the containers, handle the containers and report all their actions to the HC. They receive container load/unload instructions when the shuttle is located in the corresponding bay. When the shuttle is loaded and the crane high enough, it sends a message to the shuttle, which starts to move.

Local Traffic Control model

This model controls the local traffic in the rail area. It transmits convoy permissions generated by the LMS to the convoy of shuttles, it controls and moves the switches position in the swap areas, and it establishes and communicates to the shuttles the maximum speed they can acquire along the different rail area sections.

Host Computer model

The Host Computer is programmed to oversee the movement and location of containers on the wharf, as well as to co-ordinate and send the appropriate commands to the cranes.

SIMULATION EXECUTION

When a system is modelled, it is necessary to run it and obtain statistics on that system. The execution of the simulation allows inquiring on the status of the model run by tracing the state of elements and the values of the various variables during the simulation run.

The speed of the simulation can be controlled and moreover, turning off the animation (graphics), the simulation can run in off incredibly fast. This fact allows

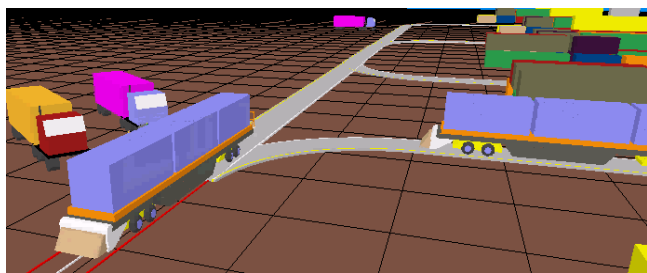


Figure 7: Part of the 3D display

during the analysis phase to run the simulation for longer duration than during the final test phase (QUEST 2001).

Figure 6 gives a general view of the ASAPPOne simulation execution. Figure 7 illustrates part of the 3D display.

CONCLUSIONS

This document describes the design and development processes carried out within the European ASAPPOne project in connection with the general operation and development of a simulator that tests the performance of the Traffic Controllers and the Line Management System. The development of a simulator has enabled the validation not only of the TCS and the LMS, but of the whole ASAPPOne concept as well.

The software design of the developed control elements is a component-based design. This, together with the simulator, has allowed to test different control techniques and select the most appropriate one.

Furthermore, the simulator in question can be used to configure other ASAPPOne models. Different models of rail areas can be stored in a DataBase, giving the possibility to test different layout configurations. The number of available shuttles or their speed and acceleration can also be selected. This enables to obtain the optimum configuration of an ASAPPOne system during the pre-design and development stage.

ACKNOWLEDGEMENT

The material of this paper is based upon work carried out inside the project ASAPPOne (Project n°: GRD1-2000-25132), backed in part by the European Community.

REFERENCES

- Evers, J. and Koppers, S. 1996. "Automated Guided Vehicle Traffic Control at a Container Terminal". *Transportation Research*, Vol. 30 Num.1.
- Ezquerro, J.M., Sanz, P., Berasategi, L. and Ruiz, A. 1995. "MARTHA-Mobile Autonomous Robots for Transportation and Handling Applications." *4th Congress of the AER*.
- Ezquerro J.M., Gonzalez J., Intxausti K. and Maruri L. 2000. "Simulation of a Container Transport System in a Harbour Terminal." *European Simulation Multi-Conference 2000*.
- Fantuzzi-Reggiane, Ikerlan, Noell Crane Systems, Sciro, Cybernetix, SudItalia Terminal Operator, Sita and Salerno Container Terminal. 2000. "ASAPPOne Description of Work" Proposal GRD1-2000-25132.

- Gambardella, L.M, Rizzoli, A.E. and Zaffalon, M. 1999. "Simulation and planning of an intermodal container terminal", *Simulation*, Vol.71 No. 2.
- QUEST. 2001. *Users Manual*. Delmia Solutions.

Scheduling and Program Agents Algorithms For E-Logistics

Professor Anatoly Levchenkovs
MSc Nadezhda Kunicina
 Transport Logistics and Intelligent Systems,
 Faculty of Transport and Mechanical Engineering,
 Riga Technical University,
 1, Kalku Street
 Riga, LV-1658, Latvia
 (+371) 7324362, (+371) 782 0094
 levas@latnet.lv, kunicina@latnet.lv,

KEYWORDS

Supply chains and logistics, e-logistics, program agents, planning and design of E-logistic networks.

ABSTRACT

The purpose of this paper is to demonstrate the algorithm for program agent, solving the problem of planning a supply chain in E-logistics.

The problem of modelling and planning of logistic operations, with the use of Internet (E-logistics), different types of servers and different types of program agents, has been considered.

The role of Supra Program Agent for solving a logistic task and co-ordination of interaction of program agents, located on server used for data storage and processing has been described. The problem of using program agents for scheduling in E-logistics has been examined.

Algorithm for program agent that offers a five-step problem solving is described.

Numerical example and practical realization of suggested algorithm for E-commerce in the Internet environment is described.

1. INTRODUCTION

This paper deals with the matter of program agents' interaction between themselves and Supra Program agent for Supply chain problem solving for scheduling modelling for e-logistics.

2. RESEARCH TASK MATHEMATICAL FORMULATION

There will be the following designations used in this work:

Distributed data bases Dp,

distributed Web server (servers) Wd,

links Web server Wr

Given set of clients K (K1, K2, ..., Ki)

Set of program agents Pa (Pa1, Pa2, ..., Pai); Pai ∈ Pa

Set of clients' tasks KPa KPa1, KPa2, ..., KPai);

KPai ∈ KPa

Assignment problem

Given set of objects (shops)

$$S = (S_1, S_2, \dots, S_n); \quad S_i \in S; \quad i = \overline{1, n}$$

set of subjects (suppliers)

$$Q = (Q_1, Q_2, \dots, Q_m); \quad Q_i \in S; \quad i = \overline{1, m}$$

number of objects' demands and subjects' criteria

$$V > 1; \quad V \in Z$$

i-th object's evaluation according to j-th criterion

$$S_i^j, \text{ where } j = 1, 2, \dots, N$$

i-th subject's evaluation according to k-th criterion

$$Q_i^k, \text{ where } k = 1, 2, \dots, N$$

Problem of defining the shortest route

Given graph M(L, O)

set of nodes – street crossroads

$$L = (l_1, l_2, \dots, l_n); \quad l \in L$$

set of arcs O – streets – which is a subset Decart product L x L

$$O \subseteq L \times L$$

set of weights for every arc $\forall o \in O$ - length of streets

$$a(o) \in A; \quad a > 0$$

pair of nodes, between which the shortest route must be found

$$(l_g, l_h); \quad l \in L, \quad g, h \in Z;$$

Problem of defining the shortest loop route (problem of travelling salesman)

Given a number of points – chosen suppliers and a shop

$$n \in Z; \quad n > 0$$

transport distance matrix between all points

$$\eta_{ij}; \quad i = \overline{1, n}; \quad j = \overline{1, n}$$

variable

$$x_{ij} = \begin{cases} 1, & \text{transport goes from point } i \text{ to } j \\ 0 & \text{in another case} \end{cases},$$

where $i, j = 1, 2, \dots, n; \quad i \neq j$

Given object function

$$F(x) = \sum_{i=1}^n \sum_{j=1}^n \eta_{ij} x_{ij} \rightarrow \min$$

under the following constraints

$$\sum_{i=1}^n x_{ij} = 1, \quad j = 1, 2, \dots, n \quad (1)$$

$$\sum_{j=1}^n x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (2)$$

T – set of problems

$$T = (t_1, t_2, \dots, t_m); \quad t \in T,$$

$$T_i - T_j + nx_{ij} \leq n - 1; \quad i, j = 1, 2, \dots, n; \quad i \neq j \quad (3)$$

Constraint (1) determines, that transport from every point leaves only once; constraint (2) – transport arrives at every point only once; constraint (3) provides loop route, that contains n points and does not contain returns.

Open conveyor schedule

Given number of processors

$$n \in \mathbb{Z}^+$$

every problem consists of n operations, i. e. $\forall t \in T$,

operation $c_j[t]$ is carried out by processor j ;

$$C_i = (c_1[t_i], c_2[t_i], \dots, c_n[t_i]);$$

$$c_j[t_i] \in C_i; \quad i = \overline{1, m}; \quad j = \overline{1, n}$$

c duration of every operation

$$W = \{w(C_i)\}; \quad w(C_i) = \{w(c_1[t_i]),$$

$$w(c_2[t_i]), \dots, w(c_n[t_i])\},$$

$$kur \quad w(c_j[t_i]) \in w(C_j) \subseteq W; \quad \text{and total}$$

$$w(c_j[t_i]) \in \mathbb{Z}; \quad i = \overline{1, m}; \quad j = \overline{1, n};$$

directive time

$$D \in \mathbb{Z}^+.$$

The problem is to make out a schedule

$$\sigma_k : T \rightarrow \mathbb{Z}^+, 1 \leq k \leq n$$

under the following constraints

From $\sigma_k(t) > \sigma_k(t)$ follows, that

$$\sigma_k(t) \geq \sigma_k(t) + w(c_k[t])$$

$$\forall t \in T, \text{ intervals}$$

$[\sigma_k(t), \sigma_k(t) + w(c_k[t])]$ do not intersect;

$$\sigma_k(t) + w(c_k[t]) \leq D,$$

where $1 \leq k \leq n, \quad 1 \leq u \leq m$

3. ALGORITHM OF PROBLEM SOLUTION

Step 1. Customer (K) enters an inquiry for solution of this complicated problem, using interface (Wr) and information function f_{Kr} :

Step 2. Supra program agent divides this problem into several subproblems

Step 3. To solve multi-criteria assignment problem, when the customer has to choose the suppliers, and the criteria are both quality and quantity criteria, using the server (fig.1.)

Step 4. Supra agent using starts inquiry processing with interface (Wr) using information function (procedure) f_{Kr} : If he receives the answer from server he can move to the next step. If the answer is not received, he has to search it in other server; otherwise the solution does not exist.

Step 5. Server (Wd) starts receiving an answer from data base (Dp) using function (procedure) f_{dp} .

Step 6. Server (Wd) starts receiving client's task solution (Pd) using function f_{pd} .

Step 7. Server (Wd) starts sending client's task solution to the server (Wr) using function f_{dr} .

Step 8. Server (Wr) starts sending client's task solution to Supra program agent SPa that makes solution (s) analysis.

Step 9. To find the shortest routes between the suppliers and the customers. Algorithm Steps analogue 3-8 are carried out.

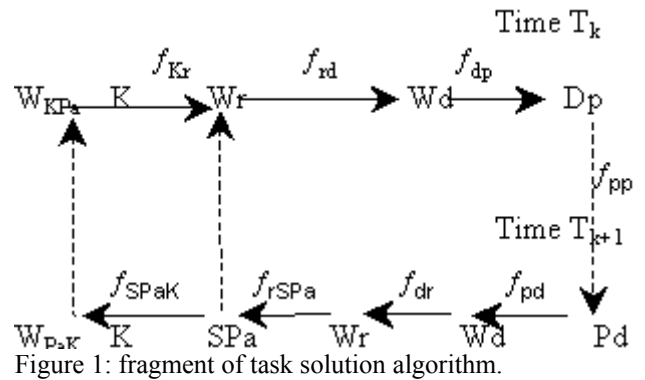
Step 10. To find the shortest loop routes between every customer and the suppliers. Algorithm Steps analogue 3-8 are carried out.

Step 11. To make up a schedule. Algorithm Steps analogue 3-8 are carried out.

Step 12. To analyze and evaluate made up schedule.

Step 13. Server (Wr) starts sending client's task solution to Supra program agent SPa that makes solution (s) analysis to offer the client only the best task solution using function f_{rSPa} .

Step 14. Supra agent (Spa) offers to the client (K) the best solution using function f_{SPaK} .



4. ROLE OF SUPRA AGENT

After solving the one logistics task is necessary to solve a new logistics problem. Supra agent coordinates the task of program agent that has to find an optimal solution for all sub problems.

5. REALISATION OF ALGORITHM

The main parts of program algorithm are similar to the algorithm above, but for computer, to perform processes that human can solve using some logical thinking, sometimes it is necessary to build very long chains of actions that involves complicated calculation procedures.

From all mentioned steps one step was not performed – the first step that considers algorithm use for multi-criteria task solution so the developed program can find the optimal supplier using one criterion.

PHP language in combination with MySQL databases is used to perform the algorithm. PHP is based on server script language that allows making dynamic Internet usage, the applications, which performance may depend on content of involved databases or results of various calculations. Web pages are made using combination of

PHP and HTML languages, but when the user opens this page, the server implements PHP commands and then sends the result to the user's Internet provider program.

Fragment of the result of assignment problem is shown in fig. 2.

	S1	S2	S3	S4	S5
G1	0	1	0	0	0
G2	1	0	0	0	0
G3	0	0	0	1	0
G4	0	0	1	0	0
G5	0	0	0	0	1

Figure 2: The result of assignment problem

After receiving solutions for assignment problem and the shortest route problem, Supra program agent can solve the problem of making up a schedule.

Let's assume that, according to optimal route of collecting goods, the suppliers are to be visited in the order S_5, S_4, S_3, S_2 , and return to the customer S_1 .

This operation must be fulfilled from 5:30 till 8:00.

Given directive time of Documentation preparing work for every i -th supplier: 5 (min).

Given directive $L_d(i)$ time of work for every i -th supplier:

$L_{d1}(5:30 - 8:00)$;

$L_{d5}(5:00 - 7:00)$;

$L_{d4}(6:00 - 10:00)$;

$L_{d3}(7:00 - 8:45)$;

$L_{d2}(7:15 - 8:00)$.

Given time of work for loading/unloading goods at the suppliers $L_p(i)$ (min):

$L_{p1}(10/10)$;

$L_{p5}(10/15)$;

$L_{p4}(05/05)$;

$L_{p3}(05/10)$;

$L_{p2}(05/15)$.

Given **time** t of delivery of goods between the suppliers $L_p(i)$ (min):

(0,1)

$L(1, 5) = 10$ min

$L(5, 4) = 15$ min

$L(4, 3) = 5$ min;

$L(3, 2) = 5$ min;

$L(2, 1) = 5$ min;

Let's use algorithm according to **EDD** rule:[13]

So, the work started at 5:30 and finished at 8:00, which is within directive period.

Supplier/time	5h	6h	7h
S_1 - Starting	5:30		
Documentation	5:30+05		
Unloading goods	5:35+10		
Loading of packaging	5:45+10		
Ending	5:55		
Transportation	5:55+10		
S_5 - Starting		6:05	
Documentation		6:05+05	
Unloading goods		6:10+15	
Loading of packaging		6:25+10	
Ending		6:35	
Transportation		6:35+15	
S_4 - Starting		6:45	
Documentation		6:45+05	
Unloading goods		6:50+05	
Loading of packaging		6:55+05	
Ending			7:00
Transportation			7:00+05
S_3 - Starting			7:05
Documentation			7:05+05
Unloading goods			7:10+10
Loading of packaging			7:20+05
Ending			7:25
Transportation			7:25+05
S_2 - Starting			7:35
Documentation			7:35+05
Unloading goods			7:40+15
Loading of packaging			7:40+05
Ending			7:45
Transportation			7:45+05
S_1 - Ending			7:50

CONCLUSION

This research work extends IT logistics possibilities. This paper offers one of possible solution of NP type tasks for program agent systems. As a result of this paper several recommendations have been worked out to make the most convenient program agents models for e logistics systems and methodology of program agents formation has been developed.

ACKNOWLEDGEMENT

The authors of this article gratefully acknowledge financial and moral support provided by Latvian Academy of Science, Riga Technical University and Railway Transport Institut .

6. REFERENCES

1. Wittig, T., 1992, "ARCHON: An Architecture for Multi – agent Systems", England.
2. Russell, Stuart J, Norvig, Peter, 1995, Artificial Intelligence: a modern approach. Prentice-Hall Canada, Inc., Toronto.
3. Ballou, Ronald H., 1992, Business Logistics Management. Prentice-Hall Canada, Inc., Toronto.
4. Turban, Efraim., 1995, Decision Support and Expert Systems. Management Support Systems. Prentice-Hall Canada, Inc., Toronto.
5. Levchenkova, A., Yansons, V., Kunicina, N., (2002), "Coordination Between Programms Agents for E-Commerce Systems", Porto, Portugal.
6. Levchenkova, A., 2001, Intelligent Logistics Transport Systems", "World Markets Series Business Briefing", European Purchasing & Supply Chain strategies, London.
7. Levchenkova, A., 2001, "Modelling for intelligent transport systems", "International conference on industrial engineering and production management", Quebec City, Canada, p. 1271 - 1276.
8. Kim, K., and Roush, F., 1987, "Them theory"; Ellis Hordwood limited, England.
9. Kunicina, N., Levchenkova, A., 2001, "Multicriterial Group Decision Making in Transportation Logistics Systems", "Modelling and Simulation 2001", Prague.
10. Polak, E., 1997, Optimization: Algorithms and consistent Approximations, Springer
11. Little J.D. An Algorithm for the Traveling Salesman Problem. Operation Research 11 (5), 972-989 ,1963
12. Jakson, M.A. Principles of Program Design. London: Academic Press, 1975.
13. Stevenson W. Production/Operations Management. Irwin. Boston. 1993

BIOGRAPHY

ANATOLY LEVCHENKOV is Associate Professor of the Logistics and Telematics at Riga Technical University from 2001. Since 1997 he has been Associate Professor at the Transport Railways Institute and from 1992 to 1997 was Head of the Riga Technical university's Department of Decision Support system. He studied Electrical Engineering in Automation and Telechanics from 1964 to 1969 and from 1969 to 1992 was a reader and associate professor at the Department of Automized Control Systems at the at Riga Technical University

NADEZHDA KUNICINA at present time is Riga Technical university, Riga, Latvia, PhD Student. She works at Riga Technical university, Transport Logistics and Intelligent Systems unit as BhD thesis supervisor. Participated in 6 International conferences, in last 3 years

ALGORITHM FOR PROGRAM AGENTS IN E-LEARNING FOR SOLVING LOGISTICS SUPPLY CHAINS PROBLEMS

Anatoly Levchenkovs
Transport and Machinery Faculty
Riga Technical University
1, Kalku St.,
LV-1658 Riga, Latvia
E-mail: levas@latnet.lv

Ilija Smyshlaev
Educational and Consulting Center
"Ric-A"
Biezina St., 7-102
LV-1029 Riga, Latvia
E-mail: s.ark@apollo.lv

KEYWORDS

Program Agent, Supply Chains, Logistics

ABSTRACT

Over the last few years rapidly developing information technologies have offered immense possibilities for individual studies, especially with the use of the Internet (E-learning). At the same time interest to logistic problems is constantly growing, including logistic supply chains problems. This paper was aimed at developing the algorithms for solving logistics supply chains problems and the problems of training exercises for students and businessmen. To achieve this the authors propose to develop and use Program Agents, which are available and can be operated any time at any server in the Internet. Practical examples of working system for the Internet are described.

INTRODUCTION

Nowadays it is very important for teachers and university staff to solve a problem of management and control of independent work of full-time, part-time and correspondent students. The problem is also urgent owing to the fact that recently, within ten years, the number of university students in some countries has nearly doubled, while the number of professorial staff has reduced nearly twice. In this paper we regard what possibilities the use of modern information technologies could provide for university staff in solving part of the problems connected with control of independent work of students in different types of education. Using of the Internet technologies for individual training of students and businessmen in solving logistics supply chains problems.

The problem is NP-solving, thus it cannot have the only possible solution.

The problem is considered by an optimality criterion including the number of unsolved training exercises.

MATHEMATICAL FORMULATION OF RESEARCH PROBLEMS

Given set of academic hours for independent work

$$H = (h_1, h_2, \dots, h_k); h \in H$$

Set of students

$$S = (s_1, s_2, \dots, s_m); s \in S$$

Set of subjects

$$P = (p_1, p_2, \dots, p_n); p \in P$$

For each $s \in S$ is given subset

$$A(s) \subseteq H,$$

which is connected with possible time of independent work for s - student

For each subject $\forall p \in P$ a subset is given

$$A(p) \subseteq H,$$

which is connected with "possible" time of independent work for p – subject at the university.

For each pair

$$(s, p) \in S \times P$$

positive integer V is assigned:

$$V(s, p) \in \mathbb{Z}_0^+$$

which is connected with "necessary" number of hours for student's independent work for every subject.

Problem: to find data function (procedure) f :

$$f: S \times P \times H \rightarrow \{0, 1\}$$

where

$$f(s, p, h) = 1$$

means that a student works on a subject p during h hours and $f(s, p, h)$ complies with the following conditions:

- 1) $f(s, p, h) = 1$ when $h \in A(s) \cap A(p)$;
- 2) for $\forall h \in H$ and $\forall s \in S$ exists such $p \in P$, when $f(s, p, h) = 1$;

- 3) for $\forall h \in H$ and $\forall p \in P$ exists such $s \in S$, when $f(s, p, h) = 1$;
- 4) for $\forall (s, p) \in S \times P$ exists such $V(s, p)$ hours h , when $f(s, p, h) = 1$;

The result, published in appropriate works*, when there is one and the only one s ,
 $s: (\exists s \rightarrow f(s, p, h) = 1)$, which complies with condition 3;
 and there is one and the only one p ,
 $p: (\exists p \rightarrow f(s, p, h) = 1)$; which complies with condition 2;

Let's supplement our problem with the following condition

- 5) for $\forall (h, s) \in H \times S$ exists such $B \subseteq P$ where for each $\forall b \in B$ satisfies $f(s, p, h) = 1$; $|P| \geq 1$
- 6) for $\forall (h, p) \in H \times P$ exists such $C \subseteq S$ where for each $\forall c \in C$ satisfies $f(s, p, h) = 1$; $|C| \geq 1$

Analogical problem with conditions [1-4] can be solved for polynomial time and it belongs to NP class of problems.

Student's time h for independent work for one lesson will be

$$|A(s)| \leq 2 \quad \forall s \in S \text{ and}$$

$$A(s) = A(p) = H, \text{ and } \forall (s, p) \in S \times P$$

ALGORITHM A OF PROBLEM SOLVING PROCEDURE

Step 1. Student in his free time h is receiving a formalized problem in subject P with interface $w_i(s) \in W$, using information function (procedure) f_{sr} : if a reply from server W_r is received, go to Step 2.

If not, take next $w_{i+1}(s)$; otherwise the solution does not exist.

Step 2. Server W_r is receiving a formalized problem in subject P from database W_d , using information function (procedure) f_{sr} .

Step 3. Server W_d is receiving a formalized problem in subject P from database D_p , using information function (procedure) f_{dp} .

Step 4. Server W_d is sending a formalized problem in subject P from database D_p , using information function (procedure) f_{pd} .

Step 5. Server W_d is sending a formalized problem in subject P to server W_r , using information function (procedure) f_{dr} .

Step 6. Server W_r is sending a formalized problem in subject P to student s with interface $w_i(s) \in W$, using information function (procedure) f_{rs} .

ALGORITHM B OF WORKING SYSTEM FOR THE INTERNET.

Step 1. Registration of a new user

At Step 1 of the algorithm Program Agent R registers new students who do not have login name, as shown in Figure 1.

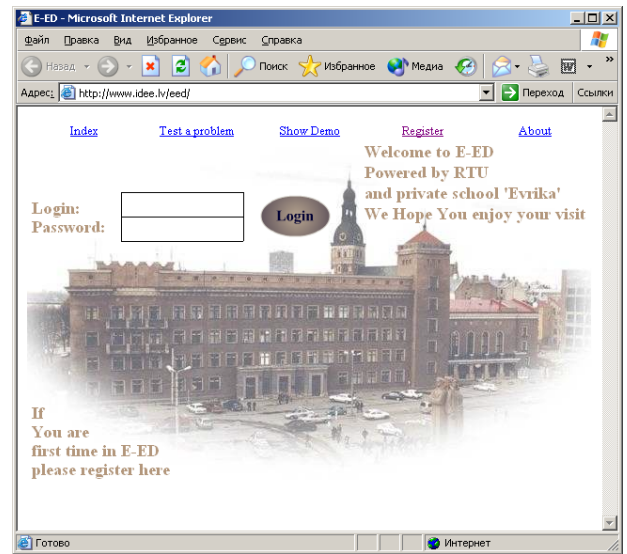


Figure 1: Registration Form for a New Student, Performed by Program Agent R .

New student registers by entering his details (login name, password, e-mail address, telephone number, post address), and gets access to E-ED, and the system additionally stores IP address of the computer in the database

Step 2. Demonstration of procedure of problem solving

At Step 2 of the algorithm Program Agent D demonstrates the procedure of solving logistics supply chains problems on a concrete example, shown in Table 1 and Figure 2.

Table 1: Fragment of Matrix of Program Agent D

Destination Town Index				
Origin Town Index	1	2	3	4
1	=	5	N/A	0
2	N/A	N/A	N/A	N/A
3	6	=	N/A	0
4	0	1	N/A	=

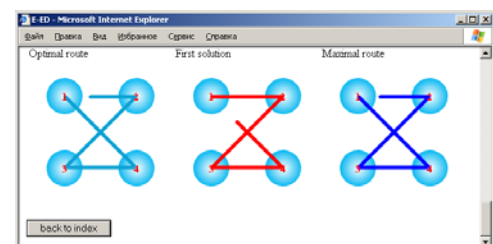


Figure 2: Graphic Example.

Step 3. Checking correctness

At Step 3 of the algorithm the student inputs data in Table 2 as shown below and his answer as shown in Table 3 Program Agent *C* checks correctness of the result received by a student for concrete training exercise.

Table 2: Input Data for Program Agent *C*.

Origin Town Index	Destination Town Index			
	1	2	3	4
1	=	6	1	3
2	4	=	1	7
3	8	2	=	4
4	3	4	3	=

Table 3: Input Answer for Program Agent *C*

Order of Passing through Towns				Total Distance
3	2	1	4	12

Student enters the matrix of input data (5x5 or 4x4) and the result, and Program Agent *C* checks correctness of the answer.

Step 4. Problem generating

At Step 4 of the algorithm Program Agent *T* offers a student formalized problem for solving. What is important that every student receives his own problem for solution and at any time can check the correctness of his results. If the result is wrong, Program Agent *T* will forward training of such student further to Program Agent *L*. Fragment of working of Program Agent *T* is shown in Table 4 and Table 5.

Table 4: Problem Generating by Using Program Agent *T*.

Origin Town Index	Destination Town Index			
	1	2	3	4
1	=	6	7	9
2	5	=	5	9
3	6	1	=	5
4	6	5	9	=

Table 5: Text for Control Work

Question No.	Text of the Problem
	Find a value of logistics activities that will be connected with the shortest route. Given towns which traveling salesman has to visit. The distances between the towns are given in the following matrix. (See Table 4 with Town Index 1, 2, 3, 4).
1.	Find a value of logistics activities that will be connected with the shortest route.
2.	Find a value of logistics activities that will be connected with the longest route.
3.	Find first solution for the shortest route

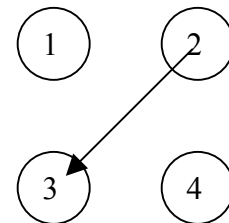
	using the algorithm, first developed by J. Little.
4.	Find second solution for the shortest route using the algorithm, first developed by J. Little.
5.	Write a minimal element for the first row.
6.	Write a minimal element for the second column.

Step 5. Learning procedure

At Step 5 of the algorithm Program Agent *L* provides three types of training: mathematical, or matrix, graphical and text. In case the student does not give correct answer to the first question of Program Agent *L*, being offered a matrix, Program Agent *L* offers him explanatory text illustrated by graphical information. This process can be repeated at any time and as many times as necessary for the student. Three types of training, provided by Program Agent *L* are demonstrated in Table 6 and Figure 3.

Table 6: Matrix #2 Step #4

Origin Town Index	Destination Town Index				C_i	A_i
	1	2	3	4		
1	=	4	N/A	0	0	4
2	N/A	N/A	N/A	N/A	0	0
3	6	=	=	N/A	0	6
4	0	0	N/A	=	0	0
Q_i	0	1	0	0	1	
B_i	6	4	0	0		



Finding A_i and B_i (minimal values in each row and column **after** transformations of matrix).

Figure 3. Matrix, graphical and text information, provided by Program Agent *L*.

Step 6. Knowledge assessment

At Step 6 of the algorithm Program Agent *M* checks the number of wrong answers given by every student at any particular step. This number is compared with maximally allowed number of wrong answers n , after that a decision on suspending of a concrete student from educational process can be made. Program Agent *M* also holds full information about every student concerning his training, including number of wrong answers, time spent on solving the problem, IP addresses from where the problem has been received and from where the solution has been sent and location of computers where training exercises have been done, for further analysis.

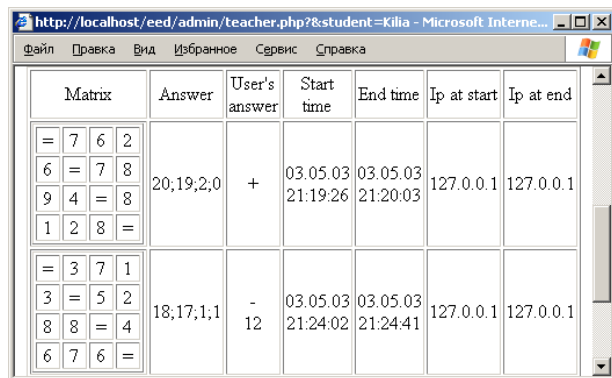
Administrator tool for the teacher is shown in Tables 7, 8 and 9.

Table 7: E-Ed Administrator Tool. Basic Information About All Students and Link to Full Information.

User-name	Mail	E-mail	Phone	IP	Number of problems solved	Number of problems failed
Kilia	Latvia, Riga, Biezina 7	kilia@inbox.lv	641114	62.85.127.1	0	1
User3	Address	u@a.lv	344550	80.18.61.18	0	0
User4	Address	a@d.lv	734475	62.85.5.227	0	0
1	1	1@1.1	154543	80.62.17.21	0	0

Table 8: Administrator Tool. Full Information About Concrete Student

User-name	Mail	E-mail	Phone	IP	Number of problems solved	Number of problems failed
Kilia	Latvia, Riga, Biezina 7	kilia@inbox.lv	641114	62.85.127.1	0	1



Matrix	Answer	User's answer	Start time	End time	Ip at start	Ip at end
<div> <div>= 7 6 2</div> <div>6 = 7 8</div> <div>9 4 = 8</div> <div>1 2 8 =</div> </div>	20;19;2;0	+	03.05.03 21:19:26	03.05.03 21:20:03	127.0.0.1	127.0.0.1
<div> <div>= 3 7 1</div> <div>3 = 5 2</div> <div>8 8 = 4</div> <div>6 7 6 =</div> </div>	18;17;1;1	-	03.05.03 21:24:02	03.05.03 21:24:41	127.0.0.1	127.0.0.1

Figure 4: Administrator Tool. Full Information About Concrete Student's Progress



Figure 5: Possible menu for Program Agents.

CONCLUSION

In addition to already developed systems the algorithm allows to add new properties or functions, such as:

automatic generation of new problems or alternatives;

full automatic control of learning process;

individual approach to teaching, considering different capabilities and level of knowledge;

24-hour 7 days a week accessibility to the system;

promptness of assessment the results of students' knowledge by the system

possibility to continue studies in emergency situations (weather conditions, epidemics, e.g. SARS, civil disorders, etc.)

The program has been tested at Riga Technical University during 2003 winter examination session in Introduction in Logistics.

ACKNOWLEDGEMENT

The authors of this article gratefully acknowledge financial and moral support provided by Latvian Academy of Science, Riga Technical University, Administration of Riga Secondary Private School "Evrika" and Dr. Yacov Pliner, the Chairman of Foundation "Parents for Children".

REFERENCES

- Jonson, J, Wood D., Wardlow D., Murphy P. Contemporary Logistics. Prentice Hall. NJ.2002.
- Russell, S.,Norvig,P. Artificial Intelligence: a modern approach. Prentice-Hall Canada, Inc., Toronto.1995
- Phillips,D.Fundamentals of network analysys. Prentice Hall. London.1981.

BIOGRAPHY

ANATOLY LEVCHENKOV is Associate Professor of the Logistics and Telematics at Riga Technical University from 2001. Since 1997 he has been Associate Professor at the Transport Railways Institute and from 1992 to 1997 was the Head of the Riga Technical University's Department of Decision Support System. He studied Electrical Engineering in Automation and Telemechanics from 1964 to 1969 and from 1969 to 1992 was a reader and Associate Professor at the Department of Automated Control Systems at Riga Technical University

ILIA SMYSHLAIEV at present studies at Riga Secondary Private School "Evrika", Riga, Latvia. Simultaneously he studies computer science at Educational and Consulting Centre "Ric-A", Riga. Participated in Annual Student Scientific Conference in Riga Technical University in 2003.

Logistics on the Factory Floor

OPTIMIZING THE SPARE PARTS PROVISIONING PLAN FOR RADAR SYSTEMS: AN APPROACH THROUGH PROCESS SIMULATION

Marco Bassanini
Francesco Costantino
Giulio Di Gravio
Ludovico Honorati

Department of Mechanical and Aeronautical Engineering
University of Rome "La Sapienza"
Via Eudossiana 18, 00184 - Rome, Italy
E-mail: francesco.costantino@uniroma1.it

KEYWORDS:

logistic simulation, radar systems, provisioning plan, system reliability, stores dimensioning

ABSTRACT

Logistics plays an important role in the life cycle of a radar system as it heavily affects the product availability and the global costs borne by the End User.

The activities required to reach an efficient and effective logistic support can be divided in two areas, the first concerning the need to affect the product design process and the second concerning the identification and management of the support resources.

One of the main activities in the latter area is provisioning, that is providing the product with the spare parts required to support it during its life.

The present article deals with the optimization of the scheduling of a spare parts provisioning plan for a given number of systems whose delivery is spread on a given time window.

The problem was approached through process simulation modeling since analytical methods turned out to be almost impossible to solve in a reasonable time for a large number of systems.

The simulation software used was Witness (rel.9.10).

The study was conducted within a collaboration between Oerlikon-Contraves Italia S.p.A. and 'La Sapienza' University of Rome.

1. INTRODUCTION

The costs related to spare parts provisioning for radar systems have a relevant influence on the overall investment for the end user.

Determining a proper spare parts provisioning time schedule is important for the following reasons:

- spread the investment for spare parts over the years, reducing the immobilized capital;
- quantify such investment during the systems' life;
- reduce the average time the spare parts spend in the arsenal without being used.

The latter point is critical especially for items that feature a limited storage life (Printed Circuit Boards with embedded batteries as an example).

The spare parts calculation for radar systems is usually

approached using rare-event statistic distributions to analytically determine the number of spares required to reach a given level of support efficacy or efficiency.

One of the parameters used to estimate the support efficacy is the stockout probability, that is the probability of witnessing a stockout for a given item, in a given time frame and with a given number of spare parts.

The stockout probability affects heavily the operational availability of the system(s), since usually a stockout results in the system being inoperative until a new spare is provided.

The efficacy of the support may be estimated through the ratio between the system operational availability and the overall support costs. Such costs should also include the money loss related to the system unavailability (\$/hour).

In the present article only the stockout probability will be considered in the discussion, neglecting both the system availability and efficiency issues.

The first part of the paper illustrates the analytical methods commonly used for calculating spare parts and the way in which such methods can be exploited to generate provisioning plans.

The second part deals with the modeling of the problem through process simulation and illustrates the results obtained.

2. ANALYTICAL METHODS

The analytical methods normally used to approach spare parts provisioning for radar systems rely on the Poisson distribution:

$$p(k) = \frac{\mu^k e^{-\mu}}{k!} \quad (2.1)$$

For spare parts calculations, the term μ is substituted with $\lambda \cdot t$, being λ the item's failure rate (failures per mega hours) and t the provisioning time horizon.

The stockout probability for a not repairable item is given by:

$$p(Q, \lambda, t) = 1 - \sum_{k=0}^Q \frac{(\lambda \cdot t)^k e^{-\lambda \cdot t}}{k!} \quad (2.2)$$

The stockout probability for repairable items may be approximated with the following expression:

$$p(Q, \lambda, t, TAT) = e^{-\lambda \cdot C(t-TAT) - \lambda \cdot TAT} \sum_{k=0}^Q \frac{(\lambda \cdot TAT)^k}{k!}$$

(2.3)

where:

$$C = \frac{(\lambda \cdot TAT)^Q}{Q!} \sum_{k=0}^Q \frac{(\lambda \cdot TAT)^k}{k!} \quad (2.4)$$

No analytical solution exists to compute the stockout probability for repairable items with a number of spares bigger than one (see [5]).

In the expressions above Q is the number of spare parts and TAT is the Turn Around Time, that is the time it takes for the repair process to be completed.

If n represents the number of items of a same kind installed on a system and N the number of systems, λ must be replaced with $\lambda \cdot n \cdot N$. Considering also the system duty cycle and the attrition rate during not-operative time, λ must be replaced with the following expression:

$$\lambda'(\lambda, n, N, \delta, k) = \lambda \cdot n \cdot N \cdot [\delta + k \cdot (1 - \delta)] \quad (2.5)$$

where δ is the system duty cycle and k the attrition factor.

2.1. Spare parts provisioning plan for one or more systems starting their duty simultaneously

If N systems start their duty at the same time, the spares provisioning plan for a given item with an assigned maximum stockout probability can be easily obtained calculating the least number of spares Q so to verify, at any time t, the condition:

$$p(Q, \lambda', t, n, N) \leq SP\% \quad (2.6)$$

if the item is not repairable or:

$$p(Q, \lambda', t, TAT, n, N) \leq SP\% \quad (2.7)$$

if the item is repairable.

The first spare parts will have to be provided at the time $t=t_1$ such that the condition above is no longer verified with $k=0$. The second spare part will have to be provided at the time $t=t_2$ such that the condition is no longer verified with $k=1$ and so on.

2.2. Spare parts provisioning plan for N systems starting their duty at different times

The problem of analytically determining the spares provisioning plan becomes more complicated if the systems are delivered not at the same time but with a given time schedule.

For not repairable items the condition (2.7) becomes:

$$1 - SP\% \leq \sum_{k_1, \dots, k_N} \left[\prod_{i=1}^N f(\lambda', t, t_i, k_i) \right] \quad (2.8)$$

where:

$$f(\lambda', t, t_i, k_i) = \begin{cases} 1 \rightarrow t < t_i, k_i = 0 \\ 0 \rightarrow t < t_i, k_i > 0 \\ \frac{[\lambda'(t - t_i)]^{k_i} \cdot e^{-\lambda'(t - t_i)}}{k_i!} \rightarrow t \geq t_i \end{cases} \quad (2.9)$$

t_i being the delivery time of the i-th system.

As an example, to calculate the provisioning time for the first spare part of an item installed on two systems starting their duty at $t=0$ and $t=t_1$ with a stock confidence level of at least 95%, the following expression must be solved:

$$e^{-\lambda' t} \cdot f(\lambda', t, t_1, 0) + \lambda' \cdot t \cdot e^{-\lambda' t} \cdot f(\lambda', t, t_1, 0) + e^{-\lambda' t} \cdot f(\lambda', t, t_1, 1) = 0.95 \quad (2.10)$$

Figure 1 shows the provisioning plan and the stockout probability curve for an item featuring a failure rate of 40 failures per million hours, duty cycle 35% with a time horizon of ten years. The first system the item is installed on is activated at $t=0$, the second after one year.

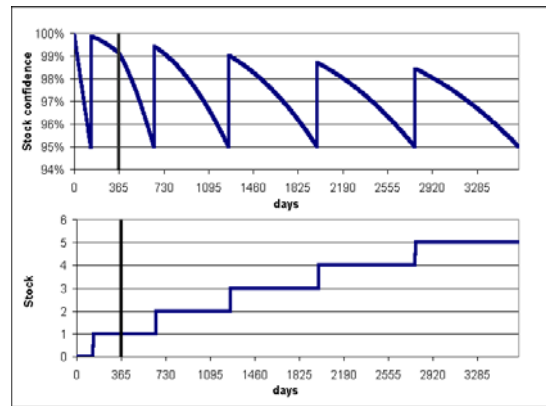


Figure 1: Provisioning plan with a stockout probability of 5%

No analytical expression can instead be obtained for repairable items since in such case the event <repaired item restocked> must be considered along with the event <item failure>, so that using the expression (2.8) exchanging the Poisson formula in (2.9) with (2.3) would not be correct.

Therefore no spare parts provisioning plan can be calculated for repairable items as it can not be related to the stockout probability.

The problem is made worse by the fact that repairable items are usually much more expensive than not repairable ones. Furthermore, even for not repairable items the expression (2.8) becomes almost impossible to solve in a reasonable time when the number of systems is large because of the time it takes to compute all the terms generated by the sum in the second term of the equation.

The only feasible solution to this problem is using Montecarlo simulation techniques and process simulation, as explained in the following paragraphs.

3. MODEL DESCRIPTION

Witness is a process simulation software with a wide spectrum of applications. Almost any type of physical or logical process can be modeled by using a set of *elements*, each one representing a specific function (such as processing, transportation, etc.).

The model discussed in the present article was developed starting from the simple concept of simulating the event "failure of the n -th component of the radar system" with the event "end of the working cycle of a *machine* element. The model scheme is depicted in figure 3.1.

All components (and relevant spare parts) were represented through a single *part* element using attributes to characterize it. Each radar system was modeled by means of multiple-instance machine and buffer elements, the i -th instance modeling the processes related to the i -th component.

When the simulation is started, the model self-sizes itself on the grounds of the information contained in an input data file, so that if n is the number of items the number of instances for the multiple-instance elements is set to n .

As mentioned above, to model the failure of the components the i -th instance of a multiple-instance machine (SYS in figure) element is assigned a negative exponential cycle time with a mean equal to the MTBF (Mean Time Between Failures) figure of the i -th component.

Limited life elements were modeled taking into account both the item's life and the item's failure rate.

Particular care was taken when selecting the pseudo-random numbers used to sample from the different statistic distributions, since this heavily affects the results of the simulation.

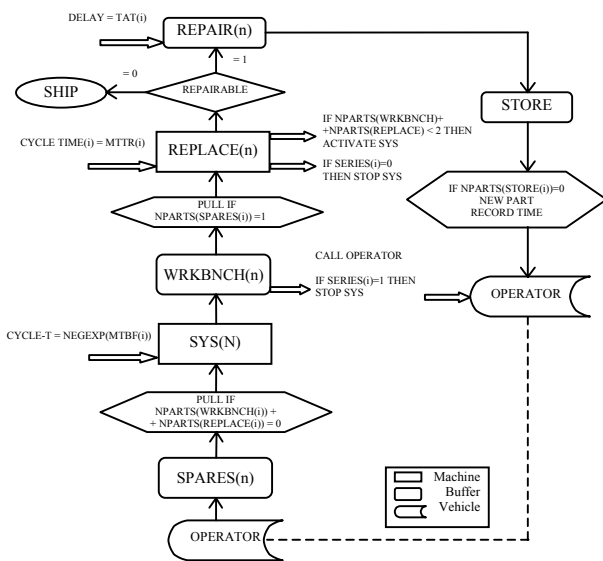


Figure 2: Block description of the model's core

When the i -th machine ends its cycle (that is, when the i -th item fails), the part is pushed to the i -th instance of a buffer element (WRKBNCH in figure), and the state of the model is modified according to the specified failure consequence (local failure or system failure).

The transport element simulating the repair team is also activated. Three type of repair teams were modeled, that depending on the failed item pick the spare from either the local or the centralized stores.

The transport element then unloads the spare in a multiple-instance buffer element (SPARES in figure). Such event triggers the repair process simulated through a machine element (REPLACE in figure), whose cycle time is set to the MTTR (Mean Time To Repair) of the failed item.

When the repair process is finished the model state is updated (SYS element is re-activated) and the failed parts are sent either to the repair process if repairable or discarded if not repairable.

The repair process was simulated through a delay buffer element, whose delay time is set to the item's Turn Around Time (TAT). When such time has elapsed, the item is newly made available as spare part.

To gather the information required to build the spares provisioning plan in each run, the model can be started with empty stores. When a spare part is required during the simulation the model automatically 'creates' it (with no delays for the repair process) and the simulation time when such event occurs is recorded, so that at the end of N runs simulating a system with n items a $N \times n \times q$ data matrix is created, being q the maximum number of spares required for any of the n items.

By analyzing such data it is possible to obtain the spare parts provisioning plan.

A Visual Basic application was realized to automate the calculations. Two confidence levels must be provided, one relevant to the number of spares required and the second linked to the provisioning time for each spare.

4. RESULTS

The model was tested on 500 runs, hypothesizing a delivery plan of eight radar systems as shown in figure 3, with a logistic time horizon of ten years.

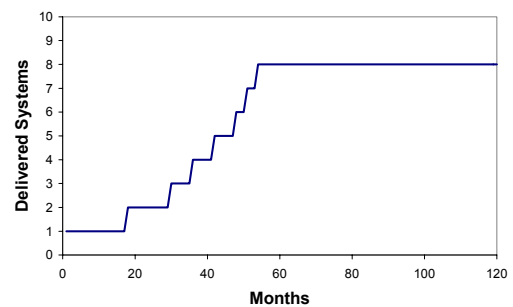


Figure 3: Delivery plan

The main goals were obtaining the spares provisioning plan and comparing the results obtained with the analytical ones. Table 4.I shows the differences in terms of number of spares required. As it can be seen, the figures are almost identical for not-repairable items. The differences for repairable items are due to the approximations in the analytical methods.

Figures 4, 5 and 6 show the spares provisioning plans obtained with different confidence levels on both the number of spares and time parameters.

Table 4.I

Confidence level	Total Stock (Simulation)	Total Stock (Analytical)	Δ for not-repairable items	Δ for repairable items
95%	225	257	-1	33
97%	246	275	2	27

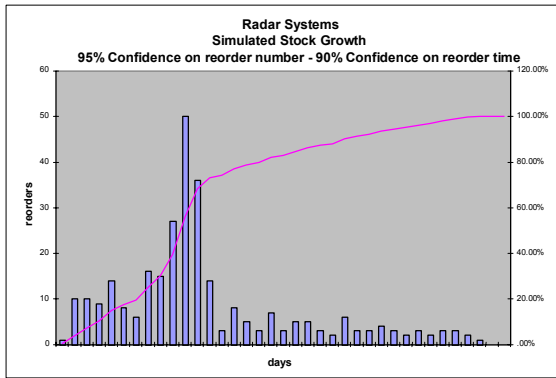


Figure 4

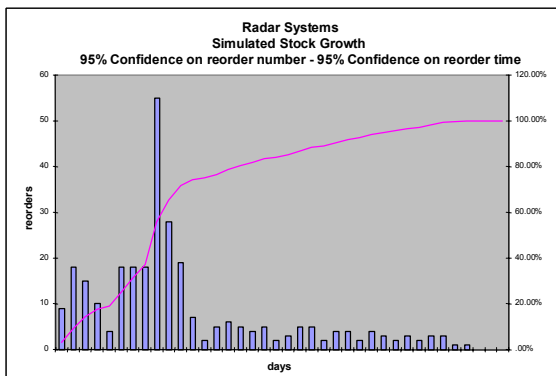


Figure 5

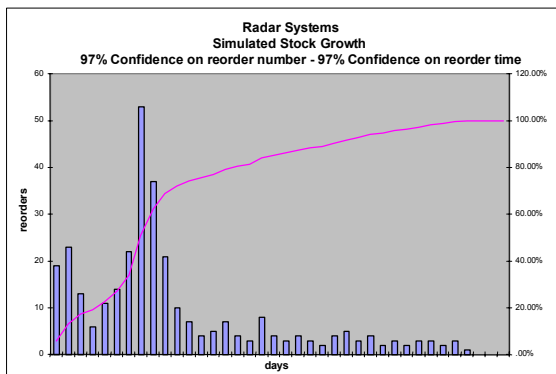


Figure 6

A comparison was made between the following two cases: in the first case the End User is provided with all the spares when the first system is shipped, while in the second case the spares are provided according to one of the provisioning plans shown above.

The economical advantage achieved when using a spares provisioning plan can be quantified by defining an interest rate on the immobilized capital and then calculating the total amount of money saved using the plan compared to immobilizing all the capital for spare parts since the beginning of the delivery plan.

In the example described below an interest rate of 3% was considered.

Since using a provisioning plan means starting the support activities with empty stores, a condition that may seem too risky, two possible approaches were considered: in the first one, the provisioning plan is followed as it is. In the

second one, an initial one-of-all spare parts buffer is provided to the End User, following the provisioning plan for the remaining quantities (the last reorder is discarded). Tables 4.II, 4.III and 4.IV below show the results obtained.

Table 4.II

95% confidence on spares amount and 90% confidence on reorder time		
Initial stock	0	1
Days of stock saved	215760	101688
Immobilized capital saved	197096	115735
Percentage on total stock value	6.8%	4%

Table 4.III

95% confidence on spares amount and 95% confidence on reorder time		
Initial stock	0	1
Days of stock saved	175490	77759
Immobilized capital saved	156817	92005
Percentage on total stock value	5.4%	3.2%

Table 4.IV

97% confidence on spares amount and 97% confidence on reorder time		
Initial stock	0	1
Days of stock saved	178537	82557
Immobilized capital saved	221858	139064
Percentage on total stock value	6.8%	4.3%

The results shown in table 4.IV as compared to the ones shown in table 4.II may seem controversial, as an increase in the confidence level of both parameters results in a higher percentage of the immobilized capital saved. Such a result can be easily explained considering that increasing the confidence on the number of spares leads to a higher investment for spares but also to a higher saving on the immobilized capital (the acquisition of the extra spares needed is added to the provisioning plan after the last scheduled acquisitions) and this factor has a stronger influence on the immobilized capital than the one related to anticipating the acquisition of the spare parts (higher confidence on reorder time).

5. CONCLUSIONS

Provisioning of spare parts is an important issue in the overall logistic support of a radar system, since it heavily affects both the availability of the supported resources and the life cycle costs borne by the end user.

All the spare parts required to support one or more systems are usually provided in a single lot with the first delivery.

This kind of approach leads to high immobilized capitals and to a high average stock age.

In the present article the advantages of using a spare parts provisioning plan and the methods to calculate it were discussed, showing the limitations of analytical methods in case of a time-spread delivery and providing an alternative solution using Montecarlo simulation.

The results obtained by simulation were validated comparing the results obtained analytically, and examples were given to quantify the advantages of using spare parts provisioning plans.

REFERENCES

- Blanchard B.S. 1974. "Logistics Engineering and Management", Prentice-Hall Inc., Englewood Cliffs, NJ.
- Birolini A. 1999. "Reliability Engineering - Theory and Practice", Springer, Berlin.
- Carrada E. 1983. "L'affidabilità per l'Elettronica" (in Italian), La Goliardica Editrice, Roma.
- Chan K.K. and M. Tortorella. 2001. "Spares-Inventory Sizing for End-to-End Availability", *Annual Reliability and Maintainability Symposium*, Philadelphia, Pennsylvania USA, January 22-25.
- Figliuzzi B. 2000. "Valutazione dell'Efficacia di Stocks di Parti di Ricambio: Modello di Affidabilità delle Scorte" (in Italian), *Rivista Tecnica Selenia*, Vol.8, n.4, April.
- Lambe C.G. 1952. "Elements of Statistics", Longmans, Green and Co Ltd, London.
- Banks J. 1998. "Handbook of Simulation, Principles, Methodology, Advances, Applications and Practice", John Wiley & Sons.

MODELING LOGIC OF SIMULATION GENERATORS FOR LEAN MANUFACTURING

Yang-Hua Lian, Hendrik Van Landeghem

Department of Industrial Management

Ghent University

Technologiepark, 903, B-9052, Ghent, Belgium

Tel: +32+(0)9-264.55.01 fax: +32+(0)9-264.58.47

email: yanghua.lian@UGent.be, hendrik.vanlandeghem@UGent.be

KEYWORDS

Simulation Generator, Simulation, Lean Manufacturing.

ABSTRACT

Globalization of the market, changing customer demands, smaller order size and the need for more product variants, force manufacturing organizations to lead their business into a new era where lean enterprise will be the norm. Lean transformation (factory layout redesigns or changes of the supply chain infrastructure) can involve high investment. Our previous research has proposed a simulation-based value stream mapping method for lean implementation. By using this implementation method, the managers can see the impacts before the physical changes and transform the organization into a lean one at minimal cost. In this paper, we aim at introducing a "simulation generator concept" which makes model building and analysis easier, powerful and also increases the value and lifecycle of the simulation models. First, we review some general ideas of lean. Then, we explain the reasons and rationales of creating simulation generators and clarify the modeling logic of the basic generators used in a lean production system such as pull demand and supermarket by flow charts and pseudo codes. The simulation generators can be reused to quickly generate different scenarios of simulation models, which help organizations to monitor and refine their real system in order to fulfill lean.

I. INTRODUCTION

"Lean" has been defined as the elimination of *muda* (waste) by Womack, Jones, and Roos (1990). Several industrial cases are illustrated in the book "Lean Thinking" (Womack and Jones 1996). In this book, crises in various business units are tackled by carrying out the 5 principles of lean philosophy: (1) define value from the perspective of the customer, (2) identify the value streams, (3) flow, (4) pull, and (5) strive to perfection.

In the lean philosophy, "value" is determined by the end customer. It means identifying what the customer is willing

to pay for, what creates "value" for him. The whole process of producing and delivering a product should be examined and optimized from the customer's point of view. So once "value" is defined, we can explore *the value stream*, being all activities – both value-added and non-value added – that are currently required to bring the product from raw material to end product to the customer. Next, wasteful steps have to be eliminated and *flow* can be introduced in the remaining value-added processes. The concept of flow is to make parts ideally one piece at a time from raw materials to finished goods and to move them one by one to the next workstation with no waiting time in between. *Pull* is the notion of producing at the rate of the demand of the customer. *Perfection* is achieved when people within the organization realize that the continuous improvement process of eliminating waste and reducing mistakes while offering what the customer actually wants becomes possible.

Tools (methods) that are part of "Lean" are addressed in literature. In contrast to the well-defined and rich set of lean tools and methods, as promoted by the Lean Enterprise Institute, there exist very few implementation methods. Our previous research has proposed a four-phases lean implementation method, which combines simulation tools and value stream mapping (Lian and Van Landeghem 2002). In phase 2 of this simulation based VSM method, we suggested to build the simulation models based on the current and future state maps of phase 1. Model building should be fairly efficient when model generators are used. *Simulation generators* are also called "*atoms*" in Enterprise Dynamics® (Incontrol Enterprise Dynamics 2001), "*templates*" in Arena ® from Systems Modeling Corp (Kelton et al. 1998), "*building blocks*" by Valentin and Verbraeck (2001, 2002) and "*simulators*" by Hamoen and Moens (2002), etc.

II. RATIONALES OF CREATING SIMULATION GENERATORS

According to Stewart (1997), information and knowledge are thermonuclear competitive weapons of our time. Knowledge is more valuable and more powerful than natural resources, big factories or fat bankrolls. Success

will come to organizations that have the best information or wield it most effectively. As a result, companies are spending more time and resources to capture information and knowledge about their dynamic flow systems. The two major challenges to gain knowledge about dynamic flow systems are (Nordgren 2001):

1. *Complexity*: With an increasing number of components in a system, it becomes more and more difficult to understand the system or to describe the relationship of the components in the system mathematically.
2. *Uncertainty*: Moreover, when uncertainty is introduced to the same system (machines breaking down, varying cycle times, changes in batch sizes or the sickness of a key employee), most analytical methods fail.

For a non-existing system (e.g. an equipment is added or relocated, process times are changed, etc.) consisting of these complexities and uncertainties, it is extremely difficult to predict how a system would perform. The only way to be able to have an insight in the logistics and dynamics of a mythical operation system is simulation. However, Valentin and Verbraeck (2002) point out that the traditional simulation studies like Banks (1998), Kelton et al. (1998), Law and Kelton (1999) start with a problem definition, make a conceptual model of the system, implement a simulation model, gather the data, perform experiments including validation and verification and analyze the outcomes, but fail to support model maintainability, the reduction of the model size and complexity, etc. Therefore, our research plan to create a library of simulation generators, which can be used to generate quickly simulation models for our lean implementation method.

As proposed by Valentin and Verbraeck (2002), a simulation generator (building block) should have following characteristics:

- Self-contained (nearly-independent)
- Interoperable (independent of underlying technology)
- Reusable
- Replaceable
- Encapsulating its internal structure
- Providing useful services or functionality to its environment through precisely interfaces
- Customizable to match any specific requirements of the environment in which it is used (plugged)

A set of our simulation generators are object-oriented modules which should aid to conduct a more effective way of simulation and contribute to the decision making processes in organizations by providing some advantages as follows:

- Reduce model size and complexity
- Reusability
- Maintainability
- Modeling time reduction

Since the simulation tool we choose is Taylor Enterprise Dynamics® (Taylor ED), from now on, we label a simulation generator as an "atom".

III. MODELLING LOGIC OF SIMULATION ATOMS USED FOR LEAN

In Taylor Enterprise Dynamics®, everything is an atom. Atoms are object-oriented model building resources. A product, a server, a queue, a model, a library and even the software itself are all atoms of Taylor ED. In addition to Taylor ED's standard atom libraries, users can create new atoms themselves. Our research aims at building a library of atoms used for lean manufacturing applications. In this section, we describe a small kanban system called "supermarket" in VSM (Figure 1), which is built using basic atoms:

1. *Demand Atom*: A customer demand is the start and the end of the supply chain. A demand (customer order) with certain inter-arrival time is the source of the flow and an order fulfillment is the end of the flow. As shown in the flow chart (Figure 1), when a demand (outside trigger) comes into the system, it requests the order quantity x_i from a supermarket atom. If there is enough kanbans (size= M and number= k , total number of parts or components in the supermarket= $M \times k$) in the supermarket atom, the demand trigger will cause a demand delivery or a machine processing before delivery (remaining kanbans= $k - x_i/M$). Otherwise, the demand will be put in a backlog queue. When considering multiple products (multiple supermarket trays), a product ID attribute is attached to the demand trigger.
2. *Supermarket Atom*: The supermarket atom is a buffer between two machine atoms (processing). It is ready to ship the parts or products when receiving a signal from a demand trigger or downstream; empty kanbans h ($h = \lceil x_i/M \rceil$, an integer) would then be released to upstream to replenish the supermarket.
3. *Machine Atom*: The machine atom receives signals from downstream supermarkets to process, ship the final products or replenish the taken parts. Each machine atom has a variable process time which is characterized by certain distribution function. In addition, a set-up time is considered in the processing machine between the fabrication of two different types of products in the case of multiple supermarket trays.

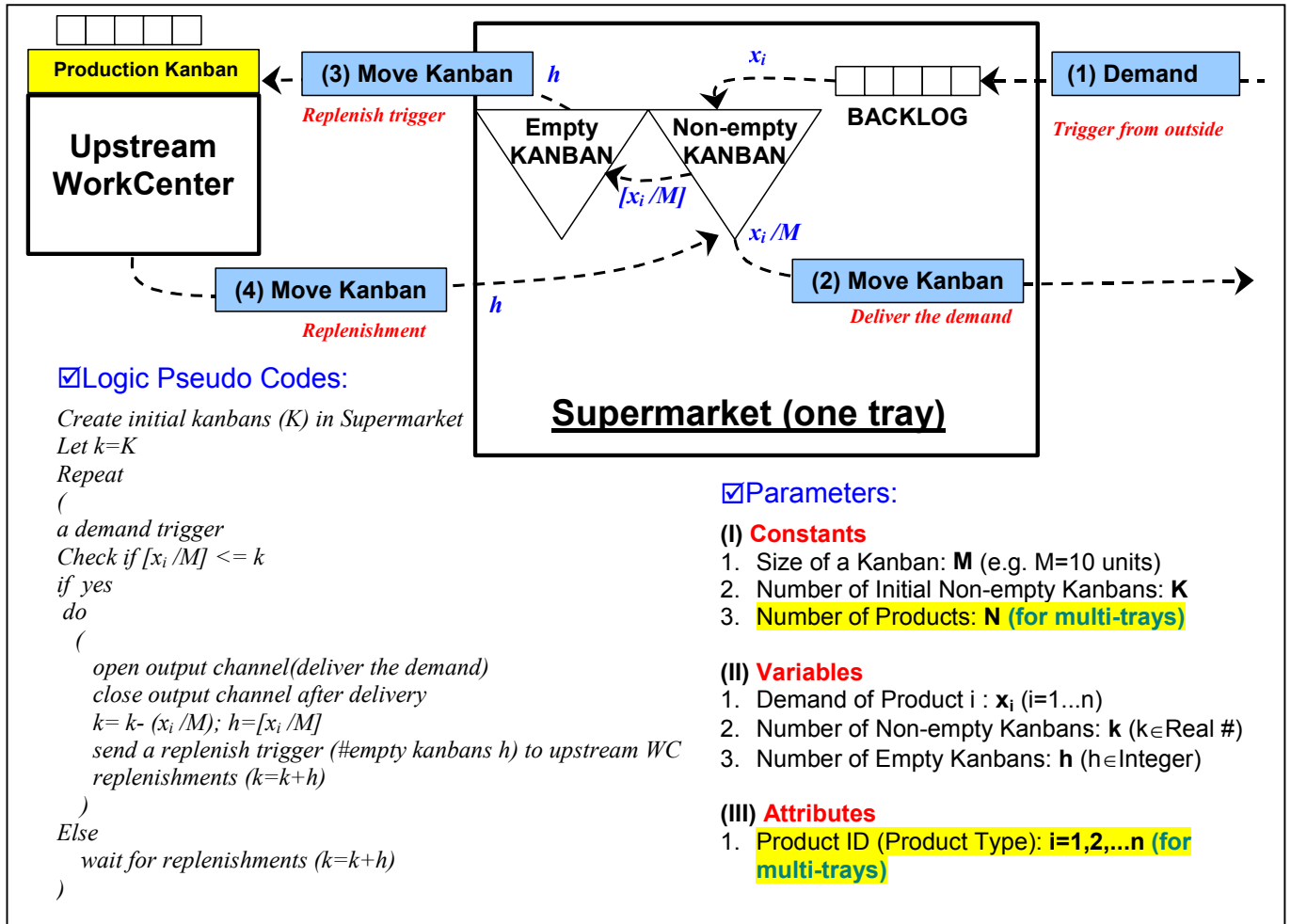


Figure 1 Atoms, Parameters and Logic Pseudo Codes in a Kanban System

We use these three basic atoms to build a small kanban system in Taylor ED (Figure 2) with assumptions: (1) the machine processing time is designed according to the demand takt time, (2) kanban size $M=1$ unit. Therefore, there is no case of backlog (because every time when there is one unit of demand trigger, one kanban is empty and the supermarket will send this empty kanban to upstream and replenish immediately). When there is no demand triggers, the supermarket atoms keep a certain inventory level which is a parameter defined by users. In Figure 2, a demand trigger comes, the machine atoms take the parts needed from supermarkets and process the parts. A material atom is built and put in the beginning of the production line to assure the supply of the raw materials always sufficient. Figure 3 is the 4D-Script codes of the supermarket atom that we built in Taylor ED.

IV.CONCLUSIONS

For the simulation based VSM method (the lean implementation method) that we propose in our previous research and supporting high complex simulation studies, we plan to build a library of atoms specially used for lean applications. In this paper, we clarify the logic of a few basic atoms that we have built and draw a flow chart, which demonstrates the relationship among them. By using these atoms, we reduce the model size and complexity during modeling processes and thus save the model building time. The models built by these reusable and extendable atoms can help an organization to monitor and create insight, knowledge about their complex and uncertain system and transform their system to a lean status.

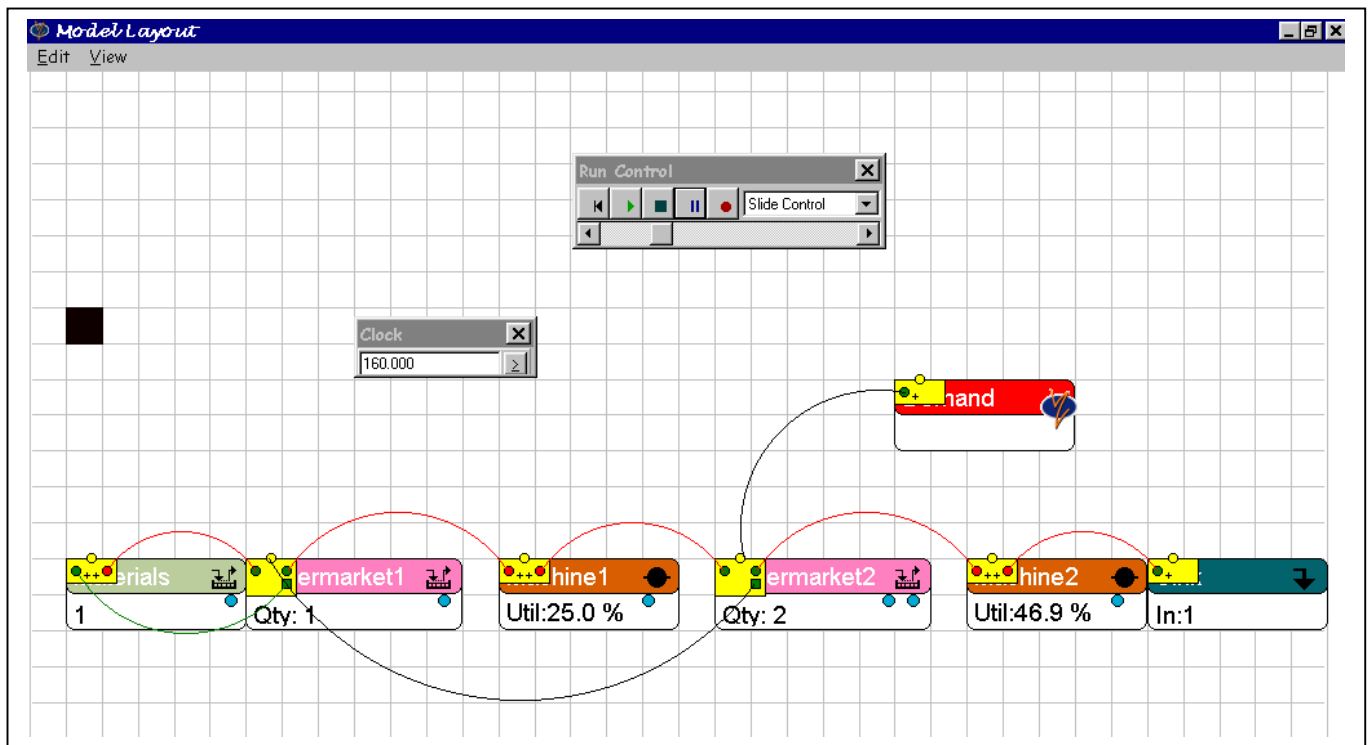


Figure 2 A 2D Simulation Scene with Demand Trigger

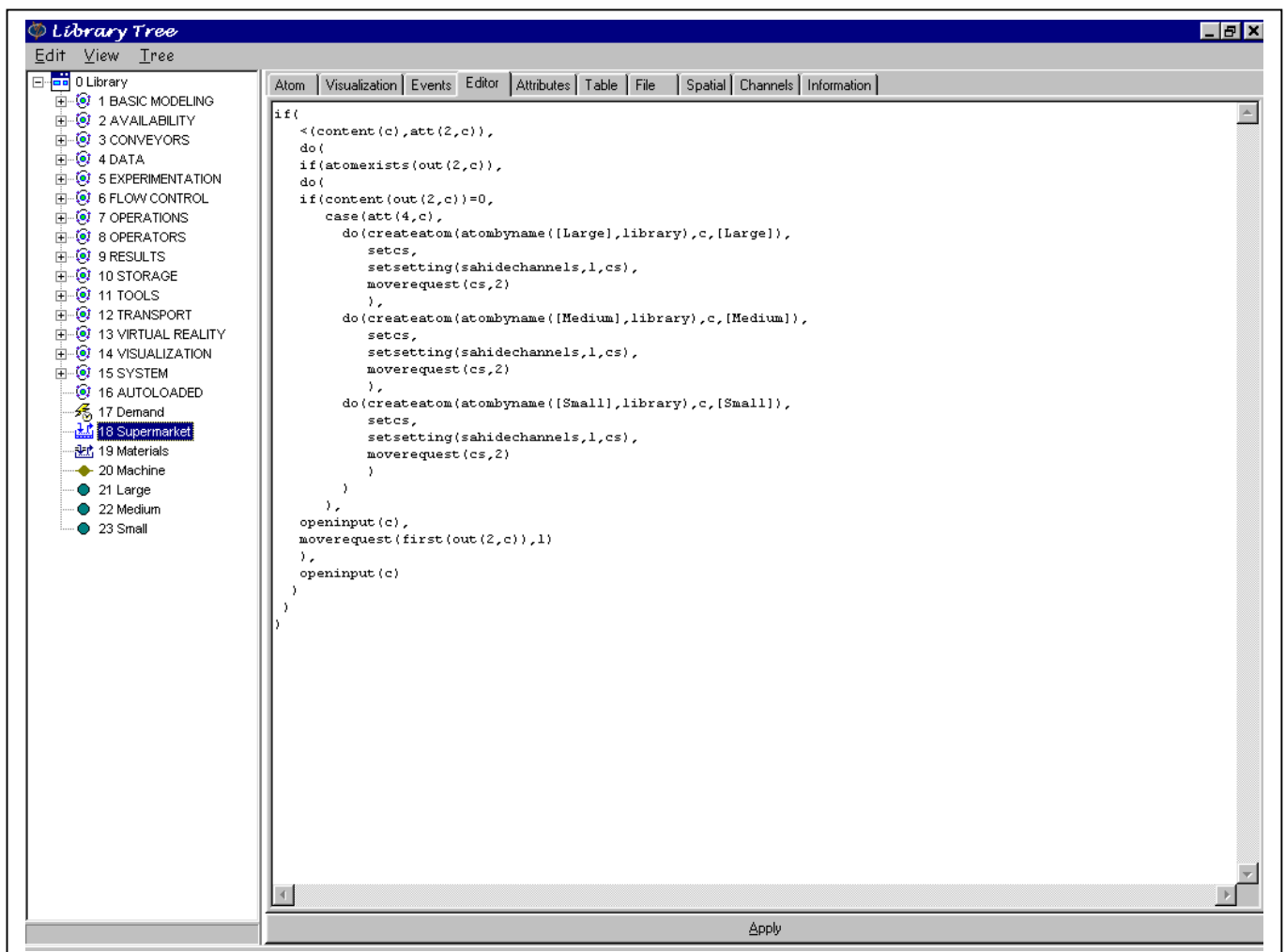


Figure 3 4D-Script Codes of the Supermarket Atom

REFERENCES

- Banks, J., 1998. *Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice*. John Wiley & Sons.
- Incontrol Enterprise Dynamics, B.V., Maarssen, The Netherlands. 2001. Enterprise Dynamics Software, *Enterprise Dynamics user manual*, 4th edition.
- Kelton, W.D.; D.A. Sadowski; and R.P. Sadowski. 1998. *Simulation with ARENA*, McGraw-Hill.
- Law, A. and W.D. Kelton, 1999. *Simulation Modeling and Analysis*, McGraw-Hill, 3rd edition.
- Lian, Y.H. and H. Van Landeghem, 2002. "An Application of Simulation and Value Stream Mapping in Lean Manufacturing", *Proceedings of 2002 European Simulation Symposium*, Dresden, Germany.
- Nordgren, W.B., 2001. "Taylor Enterprise Dynamics", *Proceedings of the 2001 Winter Simulation Conference*
- Steven, C.H. and D.J. Moens, 2002. "Logistic Simulator for Steel Producing Factories", *Proceedings of the 2002 Winter Simulation Conference*.
- Stewart, T.A., 1997. *Intellectual Capital*, 1st edition, New York: Doubleday/Currency.
- Valentin, E.C. and A. Verbraeck, 2002. "Guidelines for Designing Simulation Building Blocks", *Proceedings of the 2002 Winter Simulation Conference*.
- Valentin, E.C. and A. Verbraeck, 2001. "The Use of the Building Blocks to Enhance Flexibility and Maintainability of Simulation Models and Simulation Libraries", *Proceedings of 2001 European Simulation Symposium*, Ghent, Belgium.
- Womack, J.; D. Jones; and D. Roos. 1990. *The Machine that Changed the World – The History of Lean Production*, Harper Perennial, New York.
- Womack, J. and D. Jones. 1996. *Lean Thinking – Banish Waste and Create Wealth in your Corporation*, Simon & Schuster, New York.

based on work performed at the Georgia Institute of Technology, Atlanta (U.S.). He worked several years in industry as manufacturing consultant and in industrial informatics, as well as being Plant Manager for several years. He is now Professor at the Ghent University, as Head of the Department of Industrial Management. Rik Van Landeghem is an expert in logistic systems and their application in industry. As such he advised several companies in the choice of logistic systems and design of their logistic organization. He has written several articles on the subject. Other research interests include Concurrent Engineering. He is editor of a case book on Computer Integrated Manufacturing, as well as co-author of a book on ERP Implementations. He speaks frequently on conferences and training symposia. He is Fellow of the European Academy for Industrial Management and Member of the Institute of Industrial Engineers (IIE).

AUTHOR BIOGRAPHY

YANG-HUA LIAN was born in Taipei, Taiwan. She has a M.S. degree in Civil Engineering (majors in Transportation Engineering) from National Taiwan University (Taipei, Taiwan) and a M.S. degree in Industrial Management from Katholieke Universiteit Leuven (Belgium). From October 2001, she works as a doctoral student in the Department of Industrial Management, Ghent University (Belgium) and her research focuses on methodologies (value stream mapping, simulation, etc.) to analyse the effects of lean manufacturing.

HENDRIK VAN LANDEGHEM holds a M.S. degree in Electro-Mechanics and a M.S. degree in Industrial Management from the Ghent University (Belgium). In 1984 he obtained his Ph.D. also from the Ghent University,

ON THE WAY TO SIMULATE THE COMPUTERIZED TECHNICAL SYSTEMS

Eugene Kindler
Department of Mathematics
Ostrava University Faculty of Sciences
CZ – 701 03 Ostrava, Dvorakova 7,
Czech Republic
E-mail: evkind@ksi.mff.cuni.cz

Ivan Křivý
Department of Computer Science
Ostrava University Faculty of Sciences
CZ – 701 03 Ostrava, Dvorakova 7,
Czech Republic
E-mail: Ivan.Krivy@osu.cz

Alain Tanguy
LIMOS CNRS UMR 6158,
University Blaise Pascal, Clermont-Ferrand
Complexe scientifique des Cézeaux, F – 63177 Aubière, France
E-mail: tanguy@isima.fr

KEYWORDS

Production/logistic systems, Reflective simulation,
Nested simulation, Object-oriented programming

ABSTRACT

The paper describes ideas on modeling human-made systems using computing technique, the first stimuli and the first steps in implementation. The authors used the object-oriented programming language SIMULA for simulation of systems of that type and discovered the manner how to surmount the problems of two sorts: at one side it is to formulate exact descriptions of systems that handle with similar exact descriptions of themselves, at the other side it is necessary to respect the rules that represent a barrier for mixing both the descriptions. An automatic way to generate such models is under implementation.

INTRODUCTION – BASIC PROPERTIES OF SIMULATION PROGRAMMING TOOLS

It is possible to say that when one decides to apply computer simulation one is in a situation to solve some problems on a complex system (in case the considered system is not complex, other methods – e.g. analytical or heuristic – are commonly better than simulation). The complexity of the simulated system implies the known fact: its computer model is complex too and its programming is a hard work. The simulation languages (further SL) were designed and implemented so as to diminish the obstacles. The essence of their help consists in that their users do not need to describe “what should happen” in the applied computer (by means of an algorithm); instead of that they can describe the simulated system and this description is automatically translated into the algorithm, or – in case of some exceptional SL like the old GPSS – the description is automatically interpreted.

Although the modern SL have rich tools for optimizing or – more generally – for controlling simulation study (a sequence of simulation experiments) their kernel lies in the formalized descriptions of systems. Any description of that

type must be “natural” for the user of the given SL and therefore a SL must be oriented to some limited class of systems, for the description of which it is useful. In order to implement “tailored” simulation languages so that their translators or interpreters do not need being programmed and tested, the object-oriented programming languages (further OOPL) can serve, but the history instructed us that a small number of OOPL are suitable for that task. The main obstacle lies in the absence of quasi-parallel programming (further QPP) – although that mode of programming existed already in the very old GPSS (Schriber 1992), designed at the beginning of the sixties (Gordon 1961), although QPP was largely discussed at the world conference on SL organized by IFIP in Oslo already in 1967 (Buxton 1968) and although the first OOPL called SIMULA has admitted QPP already since 1967 (Dahl and Nygaard 1968), the popular OOPL like C++ or SmallTalk and other ones did not support QPP.

The orientation of simulation to dynamic systems implied that (simulated) time exists among the tools for system description in every SL – for some SL it serves to schedule events, for other SL it serves as an implicit entity, according which the derivatives of the functions in the differential equations are defined. Excepting some branches of modern physics, our civilization views time in the Newtonian manner and therefore the SL introduced time in this attempt, i.e. only one (simulated) time axis exists during the existence of a system and it is really an axis, i.e. the events in the system can be ordered by the trichotomy *before–contemporaneously–after*.

Similar is the attempt to the base of classes in case of simulation tools defined by help of OOPL: they offer the users a possibility to introduce any abstract concept (classes) as one for the whole computation, feeling that in every moment of the computing job, not more than maximum one simulation model can exist – following the reality they could see among us, they exclude a possibility that the same thing (a system or a component of the system) could exist more than once at the same moment.

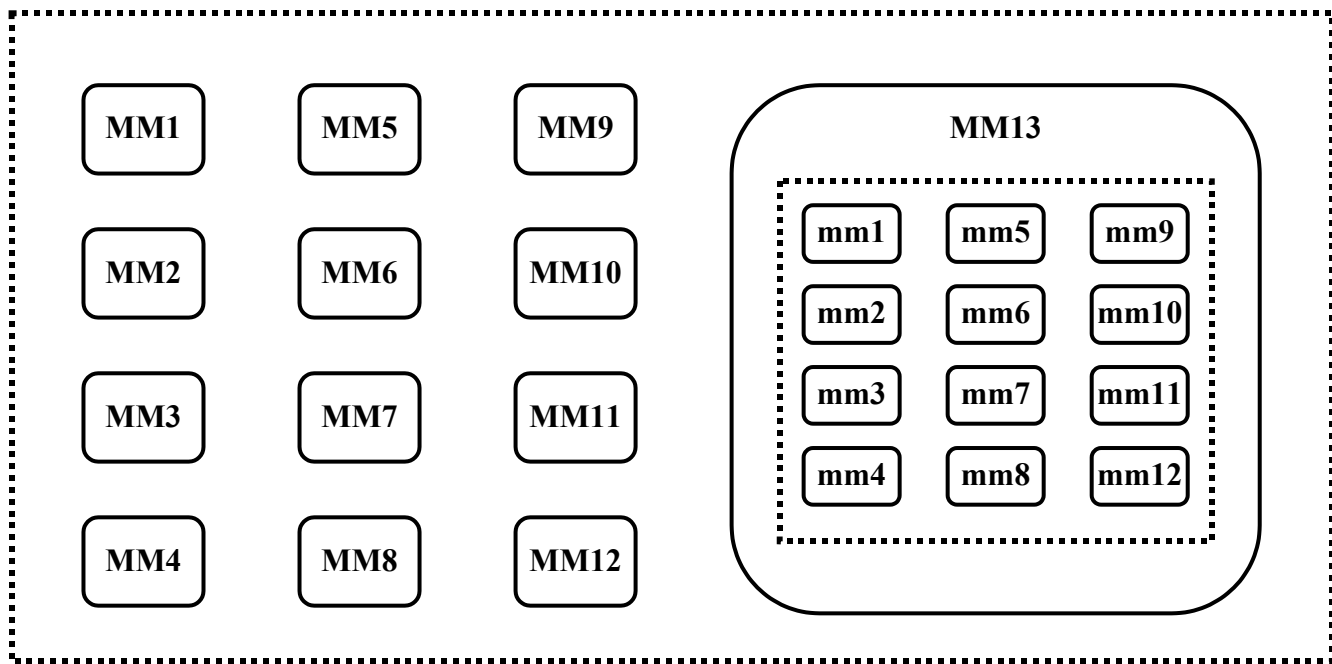


Figure 1: Man Made Objects *MMI – MM12* and Their Reflection in Another Man Made Object *MM13 – Computer*

CLASSIFICATION OF SIMULATED SYSTEMS

Let us introduce a notion of *man-made system* as a system for the function of which certain man-made components are necessary. It can be a production/logistic system, but it can be also a hospital or an environmental system. Note that man-made system does not exclude that beside *man-made components* other components exist in it, namely (parts of) nature, (organs of) humans or other living organisms etc. Let them be called *natural components*. For example, a lake itself is not a man-made system as it has only natural components, while a lake, in which fishing is made or at which a factory is placed using the water from the lake, can be viewed as a man-made system.

Habitually, almost every simulated system has certain tools that change the state of other components in a manner that we consider as physical changing – changing form, place, physical and chemical properties (e.g. temperature or velocity), changing the components existence (producing, destroying or dividing a component or unifying several components into one) etc. Let us call them *hard components*, because they must have a certain possibility to change their environment in a manner that we consider as a physical phenomenon. Beside them, so called *soft components* can be present in a system, which make decisions and govern hard components or their interactions and which are typical representatives of information processing. Note that the classifying hard/soft components is independent of that man-made/natural components, but in a system analysis it is very rare to consider a component as combined hard-soft one: it can be always considered as a pair of two components, one of which being hard and the other being soft.

In the common simulation practice existing in our civilization, we can watch two main directions. The smaller one

concerns systems in which the hard components are neglected – for example simulation of computer systems or of databases. The much greater direction concerns simulation of systems where the soft components are almost neglected or degenerated to simple activities of the hard components. This corresponds to the fact that the main interest in simulation concerns some “hard” effects of the studied systems.

COMPUTERIZED HARD SYSTEMS

Computerization is inside the horizon of the human civilization. The Europeans speak about the advent of the epoch of the “informatized” society. The consequence of the computerization or informatization is a stepwise diminishing of the number of systems with purely hard components: to them, computers and similar tools will join and will not be negligible parts of the simulated systems. Already at the end of the last century an officially expressed anticipation existed telling that “Simulation efficiently contributes to decision making either in design or in operation stages of the system life cycles” (ESMC 1999); however that does not need to be limited to simulation. One could expect to model complex system containing elements handling with complex, non-trivial models. But – similarly as in other branches – simulation could be a leading branch, which will discover the new ways for the non-simulation domains.

An example is outlined at Fig. 1, where *MMi* represent man made components; we can consider the left part of the figure as an image of a factory with its machines and other hard elements. The modern view must add one or more man made components like that represented by *MM13*. It may be a computer modeling its environment in the factory – the dotted rectangle represents such a model.



Figure 2: Reflective Simulation of a Container Yard – on the Left There is Animation of the External Model and on the Right There is Animation of the Internal Model. In it the Computer Controlling the Yard is Just Testing Whether two Paths (Represented by Trapezoids) Will not Come to a Conflict.

NESTED SIMULATION

While the phrase referenced above separated the simulation performed during the design of a system (let us call it *external simulation*) from that performed during the existence of the designed system (let us call it *internal simulation*), its consequence is clear: we must be prepared to simulate systems containing both hard elements and simulating computers: during the design – applying the external simulation – we must reflect the real existence of the designed system, i.e. the elements making the internal simulation in it.

There is a problem: to describe the model used in the external simulation (shortly: *external model*) we expect to have a suitable SL. To describe that model, we will describe the simulated system (let us call it *external system*). But in such a description, we need to describe the simulating component, i.e. the model used for the internal simulation (shortly: *internal model*) and one would wish to describe it in a usual manner: to describe the system simulated by the internal model (let us call it *internal system*). But the simulating component existing in the external model exists in a certain time (let us call it *external time*) while during that

time the internal model will be oriented to another time (so called *internal time*), modeled in the simulating component during the flow of the external time.

Therefore we need two time axes if the external model had more internal models, we would need even more than two time axes. Such a demand is not supported in the SL. We discovered that the only possibility to implement such “nested” models is to use programming languages that are object-oriented, block-oriented and agent-oriented, like Java, Beta or Simula. Using them, the following way is possible:

In any block *B*, one can use the OOPL tools and introduce a set *SIM* of classes for handling a Newtonian time flow that behaves like to exist when the computing is inside the block. The agent orientation enables the computing to enter subblocks of *B*; if the computing enters a subblock *b* into which *SIM* was also introduced, *b* behaves as having its own time axis. During that phase, the computing can handle with both time axes and when *b* is left, its time axis disappears but the time of *B* remains. That reflects the event that the simulating component finished its simulation and expects some future activity (may be another simulation).

REFLECTIVE SIMULATION

In case the simulating component is oriented to some other subject than the system in which it exists, the problem is solved. Comparing with the example presented in Fig. 1, one could imagine that the contents of the rectangle inside *MM13* does not correspond to the entities at the left of *MM13*, i.e. the interior of the rectangle is a model of something different than the environment of *MM13*. However, the most natural situation is that where both the external and internal models concern the same object, i.e. the internal system is very similar to the external one. Thus, Fig. 1 becomes a suitable metaphor for the future reality of modern computerized enterprises. Such a case is called **reflective simulation**. An obvious demand is to describe both the systems, and therefore models, using the same language. It can be realized when in both the blocks *B* and *b* the same classes are introduced. In other words, instead of *SIM*, we put a larger set of classes into both the blocks. Then we can describe in both the blocks the corresponding models.

This practice carries a danger, called **transplantation**. It consists in an erroneous mixing of components of both internal and external models, for example: in putting an element belonging to the internal model into a queue declared for the external model. It is very difficult to find the true root of such an error. JAVA permits this error. BETA checks it during the computation (so the compiled models run slowly) and SIMULA has a certain syntactical control that leads the user not to avoid the implementation of transplantation in his programs. That causes the fast run of the models written in SIMULA but during 25 years the limitation seemed to be so strong that it did not allow a communication between the internal and external model, i.e. to eliminate a practical use of SIMULA. It was not sooner than in 1993 when the tricks enabling the communication were discovered (Kindler 1993).

Nevertheless, the tricks are rather sophisticated and can be used only by persons having good knowledge of SIMULA. Therefore we turned our attention to automatic conversion of descriptions of conventional (non reflective) simulation models to description of reflective ones. In other words, one can describe a system *S* and the automatic conversion enlarges the description so that another system *S** is described, containing a class *C* of computers that are able to react at a simple signal *P* so that they watch their actual environment, represent it as initial states of models that they immediately generate and simulate what could happen in *S**. The user has only to determine the number of the instances of *C*, to place signals *P* and to describe the reason why the instances have to simulate.

CONCLUSION

The horizon of the application is as large as that of the internal simulation. Until nowadays, the reflective simulation of queuing systems and of container yards was performed. In case of queuing systems, the internal simulation anticipates the consequences of certain decisions concerning the opening/closing of facilities, for which queues arise (Kindler 1996). In case of container terminals, the internal simulation anticipated what could happen in the labyrinth among the containers and refused the traces of the ground-moving transport tools, which could lead to deadlocks (Kindler 2000). In Fig. 2, there is an illustration: the animation represents a view of the yard taken from a bird's eye, i.e. in a similar way as for a geographical map, the number of containers in the columns is represented by digits and by shadow (originally by color). As a great source of stimuli the cyclical conveyors appear, as they can anticipate a lot of future events by simulation.

The applications allowed us to discover many possibilities offered by SIMULA. One can apply them already when

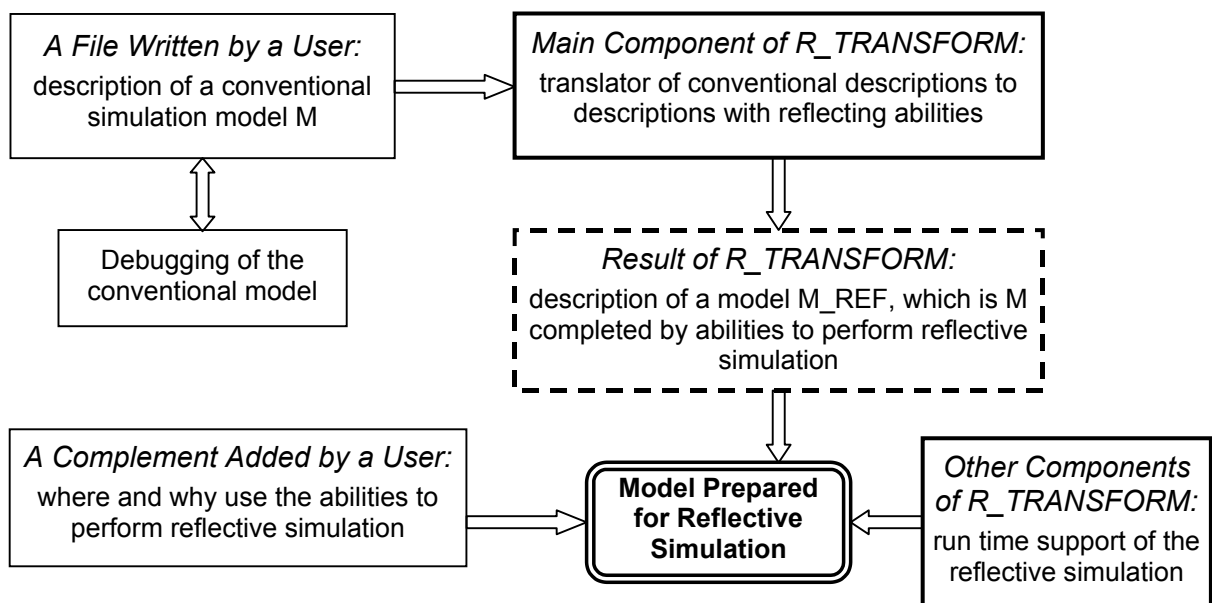


Figure 3: Scheme of R_TRANSFORM Work

formulating a conventional simulation. The plenty creates a large spectrum of demands for the automatic conversion mentioned at the end of the preceding section, but the works concerning its implementation continue (Kindler et al. 2002). The result is a software product called R_TRANSFORM and its basic structure is outlined at Fig. 3. The components with bold edges are either parts of R_TRANSFORM or products of R_TRANSFORM. Note that in the general case it is not possible to algorithmize the process how the users come to their demands concerning the duration of the internal simulation, the conditions, under which the internal simulation should be activated, and the use of the data generated by the internal simulation. Such demands are to be formulated by the users.

ACKNOWLEDGEMENT

The paper is one of the results of an international teamwork activity performed in the frame of project Barrande between French and Czech universities.

REFERENCES

- Schriber, T. J. 1990. *An Introduction to Simulation Using GPSS/H*. Wiley, New York – Chichester – Brisbane – Toronto – Singapore
- Gordon, G. 1961. "A general purpose systems simulation program". In *Proceedings 1961 EJCC*. MacMillan, New York, 1-24
- Buxton, J. N. (Ed.). 1968. *Simulation Programming Languages – Proceedings of the IFIP Conference on Simulation Programming Languages Held in Oslo, May 1967*. North Holland, Amsterdam
- Dahl, O.-J. and K. Nygaard. 1968. "Class and Subclass Declarations", in *Simulation Programming Languages – Proceedings of the IFIP Conference on Simulation Programming Languages Held in Oslo, May 1967*, J. N. Buxton (Ed.), North Holland, Amsterdam, 158-174
- ESMC 1999. *First (electronic) announcement of the European Simulation Multiconference 1999 (Warsaw)*, Society for Computer Simulation, Ghent
- Kindler, E. 1993. "Reflective simulation in SIMULA". In: *Applications of Distributed and Graphical Simulation – Proceedings of the 19th Conference of the ASU*, Aberdeen, Kings College (University of Aberdeen), D-2-1 – D-2-11
- Kindler, E. 1996. "Reflective Simulation (First Experiences)". In: *Simulation und Animation für Planung, Bildung und Präsentation '96*. [ASIM Mitteilungen, Heft Nr. 54], ASIM, Magdeburg – Wien, 39-50
- Kindler, E. 2000. "Nesting Simulation of a Container Terminal Operating With its own Simulation Model". *JORBEL (Belgian Journal of Operations Research, Statistics and Computer Sciences)*, 40, No. 3-4, 169-181
- Kindler, E., I. Krivy, and A. Tanguy 2002. "Reflective Simulation of Logistic and Production Systems". In *International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation HMS, Modelling & Applied Simulation MAS 2002, October 3-5, 2002, Begeggi, Italy*, A. G. Bruzzone, Z. Merkuryev and R. Mosca (Eds.). DIP (Department of Production Engineering) – Genoa University, Liophant Simulation Club, McLeod Institute of Simulation Science – Genoa Center, Genoa, 97-102

EUGENE KINDLER studied at Charles University in Prague where he got a PhD (doctor of philosophy) degree

in logic and a RNDr (doctor of sciences) degree in mathematics. He worked as research specialist with Research Institute of Mathematical Machines in Prague (1958-66), and then as scientific specialist with Biophysical Institute at the Faculty of General Medicine of Charles University (1966-73) and with Faculty of Mathematics and Physics (1974-2000) of the same University. He was visiting professor with the University at Pisa (Italy), at Morgantown (West Virginia), at Clermont-Ferrand and at Lorient (France) and foreign lector of Humboldt University (Berlin). Czechoslovak Academy of Science gave him a degree CSc (Candidate of sciences) in physics and mathematics. Nowadays he is associate professor at a new University of Ostrava in Czech Republic. He is a member of Liophant Simulation Club and since 1993 he was a senior member of The Society for Computer Simulation. E. Kindler was the author of the first Czechoslovak ALGOL compiler and of the first Czechoslovak digital simulation system. Since 1967, his main interests have been oriented to simulation, the object-oriented programming and to applications in industry, life sciences and transportation.

After having finished his study at the Czech Technical University in Prague, **IVAN KRIVÝ** worked for 13 years at the Nuclear Research Institute (1962-1975) and later (1975-1977) he worked at the Computer Art Establishment as a system programmer. In 1974 he graduated at the Charles University and in 1975 he obtained the degree Ph.D. in solid state physics. Since 1977 he was teaching courses in programming, probability theory and statistics at the Pedagogical Faculty at Ostrava. After the University of Ostrava having been established in 1991, he entered the Mathematics Department of the Faculty of Science and nowadays he works in the Computer Science Department as Professor in applied mathematics. Since 1992 his research activities are oriented to dynamic modelling and computer simulation (e. g. local epidemics, population dynamics) as well as to stochastic algorithms and their use in the global optimization. I. Krivý is a member of the American Mathematical Society, the Society for Industrial and Applied Mathematics, the Association of Simula Users, and the International Association for Statistical Computing. He is author of about 160 publications including 4 monographs. Stays abroad: Silesian University of Katowice (1991), Portsmouth Polytechnic (1991), University of Wrocław (1995), Universidad de Córdoba (1996, 2000), and Lulea University of Technology (1999) and Blaise Pascal University of Clermont-Ferrand.

ALAIN TANGUY studied mathematics and computer sciences at the French University Blaise Pascal of Clermont-Ferrand. He received a DEA in real world modeling and a Ph.D. in modeling and simulation of manufacturing systems. He has taught as assistant professor at the CUST engineering high school and presently at ISIMA as MCF. Since his graduation he has worked as researcher at the University laboratory LIMOS. His main research interests include object-oriented modeling with process-oriented approach and evaluation using mathematical methods and discrete event simulation of systems providing a service, such as manufacturing or transportation systems. Since 1995 he has been chairman of the Association of Simula Users.

Logistics and Traffic Simulation

THE USE OF AGGREGATE APPROACH FOR SIMULATION OF LOGISTICS SYSTEMS

Henrikas Pranevicius
Business Informatics Department,
Kaunas University of Technology
Studentu 56-301, LT-3028, Kaunas, Lithuania
E-mail: hepran@if.ktu.lt

KEYWORDS

Aggregate approach, logistics system, simulation model, formal specification, validation.

ABSTRACT

The paper presents simulation model of a logistics system that has been created using aggregate approach that permits to integrate behavior and performance analysis of the base of single specification. Simulated system is formed of shopping centers, warehouse and procedures. Logistics system model was validated using reachable state approach that permits to analyze such system properties: boundedness, absence of over specification, completeness, static and dynamic deadlocks freeness, and termination. Model permits to evaluate such characteristics of the system: mean quantities of goods and their shortages in shopping centers and warehouse.

INTRODUCTION

Logistics system is one of the most important component of business process that is used for control of product transportation flows, product warehousing, stock supply and manufacturing. It is impossible to establish effective business system without evaluation of logistics cases. While organizing trade or manufacturing processes competitive ability of market participant as well as its rising in the market depends on reasoned distribution of goods, stocks and manufacturing resources.

Logistics systems belong to class of distributed systems (Lynch 1996). Two properties are essential for distributed systems:

- computation activity is represented as the concurrent execution of sequential processes,
- processes communicate by passing messages.

The models of computation generally considered to be distributed are process models, in which computational activity is represented as the concurrent execution of sequential processes. The process models that are most obviously distributed are ones in which process communicate by message passing: one process sends a message by adding it to a message queue, and another process receives the message, by removing it from queue. These models vary in such details as the length of the

message queues and how long a delay may occur between when a message is sent and when it can be received.

Two kind of analysis are used for analysis of distributed systems: behavior and performance analysis. All possible trajectories are analyzed during behavior analysis and it permits to check a correctness of specification. Various validation and verification methods are used for correctness analysis. During performance analysis developed specification of distributed system is executed by computer. Performance analysis is carried out by simulation means.

Popular specification languages used for design of distributed systems are:

- in protocol engineering: SDL, ESTELLE, LOTOS;
- in software engineering: Z, VDM, Raise.

Mathematical methods used in specification languages are:

- extended state automata: SDL, ESTELLE;
- calculus of communicating systems: LOTOS;
- theory of sets and mathematical logic: Z, VDM;
- piece-linear aggregates (piece-linear Markov process): ESTELLE/Ag.

The suggested in the paper technique is tightly related to aggregate approach, which permits to integrate behavior and performance analysis on the base of single specification. Such possibility permits to prove that developed specification is correct and to evaluate performance characteristics of an analyzed system.

N.Buslenko initially proposed aggregate approach for creation of simulation model (Buslenko 1971). Author of this paper extended this approach by introducing controlling sequences (Pranevicius 1991). It permitted to use this mathematical scheme for formal specification and analysis of complex systems.

Due to rising competition it is important to provide effective supply of goods and avoid its lack as well as to reduce expenses related to keeping of goods remainder. Thus one of the important task of logistics is to determine how much, when and where to ship goods as well as how much and when to order them from the supplier. This task can be resolved using respective mathematical models (Daganzo 1999; Verbaeck et al. 2000; Bruzzone et al. 2000; Bruzzone and Mosca 1999).

Paper presents model of logistics system that was created using Aggregate method (Pranevicius 1992; Pranevicius 2000). In the paper aggregate specification of logistics center is presented that was used creating system simulation model. PRANAS-2 (Pranevicius 1994) system was used for model software implementation.

AGGREGATE SPECIFICATION OF A COMMERCE LOGISTICS CENTRE

Conceptual model. System consists of N shopping centres, Warehouse and Producers. The request of goods from Shopping centre is passed to Warehouse when an amount of goods in Shopping centre becomes less than threshold size. Each request in Warehouse is processed. If amount of goods in Warehouse is enough, then goods are delivered to Shopping centre. In other case request is placed to queue of requests. The functioning of Warehouse is the same as Shopping centre.

Aggregate specification consists of aggregates Shopping_centre_i, $i = \overline{1, N}$; Warehouse and Producers (Fig.1).

Aggregate Shopping centre_k, $k = \overline{1, N}$.

1. *The set of input signals.* $X = \{x_1\}$, where $x_1 = (x_{11}, x_{12})$, and $x_{11} = k$, x_{12} - amount of delivered goods.
2. *The set of output signals.* $Y = \{y_1\}$, where $y_1 = (y_{11}, y_{12})$ and $y_{11} = k$, y_{12} - amount of requested goods.
3. *The set of external events.* $E' = \{e'_1\}$, where e'_1 means that goods have been delivered.
4. *The set of internal events.* $E'' = \{e''_1\}$, where e''_1 means the time instant, when amount of goods at shopping centre is evaluated.
5. *Controlling sequences.* $e''_1 \rightarrow \{x_j\}$, $j = \overline{1, \infty}$, where $\{x_j\}$ - time interval after which amount of goods at shopping centre is evaluated.
6. *Parameters.* Δs - the amount of requested goods, s_0 - threshold. When $s(t_m)$ becomes less than s_0 request for goods is issued.
7. *Discrete component of state.* $\mathbf{n}(t_m) = \{s(t_m), s_0(t_m)\}$, where $s(t_m)$ - stocks of goods in Shopping centre, $s_0(t_m)$ - value of threshold.
8. *Continuous component of state.* $z_n(t_m) = \{w(e'_1, t_m)\}$.
9. *Initial state:* $s(t_0) \geq s_0$, $w(e''_1, t_0) = t_0 + x_1$, $s(t_0) = s_0$.
10. *Transition and output operators:*

$H(e'_1)$:

$$s(t_{m+1}) = s(t_m) + x_{12}.$$

$H(e''_1)$:

$$s(t_{m+1}) = \max(0, s(t_m) - h_j),$$

where h_j - demand for goods.

$$w(e'_1, t_{m+1}) = t_m + x_j.$$

$G(e'_1)$:

$$y_1 = (k, \Delta s), \text{ if } s(t_m) \geq s_0 > s(t_{m+1}).$$

Aggregate Warehouse

1. *The set of input signals.* $X = \{x_1, x_2, \dots, x_N, x_{N+1}\}$, where $x_1 = (x_{i1}, x_{i2})$, x_{i1} - number of shop, x_{i2} - amount of requested goods, x_{N+1} - amount of goods from producers.
2. *The set of output signals.* $Y = \{y_1, y_2, \dots, y_N, y_{N+1}\}$, where $y_i = (y_{i1}, y_{i2})$ and $y_{i1} = k$, y_{i2} - amount of requested goods.
3. *The set of external events.* $E' = \{e'_1, e'_2, \dots, e'_N, e'_{N+1}\}$, where $e'_i, i = \overline{1, N+1}$ - arrival of x_i signal.
4. *The set of internal events.* $E'' = \{e''_1, e''_2, \dots, e''_N, e''_{N+1}\}$, where $e''_i, i = \overline{1, N}$ means time instant goods delivery to i -th shopping centre, e''_{N+1} means end of request processing for goods delivery.
5. *Controlling sequences.* $e''_i \rightarrow \{x_{ij}\}$, $i = \overline{1, N}$, $j = \overline{1, \infty}$, $e''_{N+1} \rightarrow \{h_j\}$, $j = \overline{1, \infty}$, where x_{ij} - duration of time interval for goods delivery to i -th shopping centre; h_j - duration of time interval for request processing.
6. *Parameters.* Δs - the amount of requested goods, s_0 - threshold.
7. *Discrete component of state:*
 $\mathbf{n}(t_m) = \{s(t_m), d_1(t_m), \dots, d_N(t_m), c(t_m), q(t_m), Q(t_m), s_0(t_m)\}$, where $s(t_m)$ - stocks of goods in Warehouse; $d_i(t_m)$ - amount of delivered goods to i -th shopping centre, $s_0(t_m)$ - value of threshold,

$$c(t_m) = \begin{cases} 1, & \text{request of goods is processed} \\ & \text{from } i\text{-th shopping centre,} \\ 0, & \text{otherwise;} \end{cases}$$
 $q(t_m)$ - amount of goods in request.
8. *Continuous component of state:*
 $z_n(t_m) = \{w(e'_1, t_m), \dots, w(e'_N, t_m), w(e'_{N+1}, t_m)\}$.
9. *Initial state.* $s(t_0) \geq s_0$, $Q(t_0) = \emptyset$, $s_0(t_0) = s_0$.
10. *Transition and output operators:*

$H(e'_1)$: $i = \overline{1, N}$.

$$s(t_{m+1}) = \begin{cases} s(t_m) - x_{i2}, & \text{if } s(t_m) > x_{i2} \wedge \\ & w(e'_{N+1}, t_m) = \infty, \\ s(t_m), & \text{otherwise;} \end{cases}$$

$$Q(t_{m+1}) = \begin{cases} Q(t_m), & \text{if } s(t_m) > x_{i2} \wedge \\ & w(e'_{N+1}, t_m) = \infty, \\ ENQ(Q(t_m), (x_{i1}, x_{i2})), & \text{otherwise;} \end{cases}$$

$$q(t_{m+1}) = \begin{cases} x_{i1}, & \text{if } s(t_m) > x_{i2} \wedge w(e'_{N+1}, t_m) = \infty, \\ q(t_m), & \text{otherwise;} \end{cases}$$

$$w(e''_{N+1}, t_{m+1}) = \begin{cases} t_m + \mathbf{h}, & \text{if } s(t_m) > x_{i2} \wedge \\ & w(e''_{N+1}, t_m) = \infty, \\ w(e''_{N+1}, t_m), & \text{otherwise;} \end{cases}$$

$G(e'_1)$:

$$y_{N+1} = \Delta s, \text{ if } s(t_m) \geq s_0 > s(t_{m+1}).$$

$H(e'_{N+1})$:

$$s(t_{m+1}) = \begin{cases} s(t_m) + x_{N+1} - n_2, & \text{if } Q(t_m) > 0 \wedge \\ & w(e''_{N+1}, t_m) = \infty \wedge \exists k > 0, \\ s(t_{m+1}) + x_{n+1}, & \text{otherwise;} \end{cases}$$

$$Q(t_{m+1}) = \begin{cases} DEQ(Q(t_m), k), & \text{if } Q(t_m) > 0 \wedge \\ & w(e''_{N+1}, t_m) = \infty \wedge \exists k > 0, \\ Q(t_m), & \text{otherwise;} \end{cases}$$

$$\mathbf{c}(t_{m+1}) = \begin{cases} n_1, & \text{if } Q(t_m) > 0 \wedge w(e''_{N+1}, t_m) = \infty \wedge \\ & \exists k > 0, \\ \mathbf{c}(t_m), & \text{otherwise;} \end{cases}$$

$$q(t_{m+1}) = \begin{cases} n_2, & \text{if } Q(t_m) > 0 \wedge w(e''_{N+1}, t_m) = \infty \wedge \\ & \exists k > 0, \\ q(t_m), & \text{otherwise.} \end{cases}$$

$$w(e''_{N+1}, t_{m+1}) = \begin{cases} t_m + n_j, & \text{if } Q(t_m) > 0 \wedge \\ & w(e''_{N+1}, t_m) = \infty \wedge \exists > 0, \\ w(e''_{N+1}, t_m), & \text{otherwise.} \end{cases}$$

$$k = \min_{1 \leq j \leq \#Q(t_m)} \left\{ j \mid Q_j(t_m) = (n_1, n_2), n_2 \leq s(t_m) + \mathbf{c}_{N+1} \right\}.$$

$G(e'_{N+1})$:

$$y_{N+1} = \Delta s, \text{ if } Q(t_m) > 0 \wedge w(e''_{N+1}, t_m) = \infty \wedge \exists k > 0 \wedge s(t_m) + \mathbf{c}_{N+1} - n_2 < s_0.$$

$H(e'_i)$: $i = \overline{1, N}$.

$$d_i(t_{m+1}) = 0.$$

$G(e''_i)$:

$$y_i = d_i(t_m).$$

$H(e'_{N+1})$:

$$d_{\mathbf{c}(t_m)}(t_{m+1}) = q(t_m),$$

$$w(e'_{\mathbf{c}(t_m)}, t_{m+1}) = t_m + \mathbf{x}_{\mathbf{c}(t_m), j},$$

$$k = \min_{1 \leq j \leq \#Q(t_m)} \{ j \mid Q_j(t_m) = (n_1, n_2), n_2 \leq s(t_m) \},$$

$$(n_1, n_2) = Q_k(t_m),$$

$$\mathbf{c}(t_{m+1}) = \begin{cases} n_1, & \text{if } \exists k > 0, \\ 0, & \text{otherwise;} \end{cases}$$

$$q(t_{m+1}) = \begin{cases} n_2, & \text{if } \exists k > 0, \\ 0, & \text{otherwise;} \end{cases}$$

$$w(e''_{N+1}, t_{m+1}) = \begin{cases} t_m + \mathbf{h}_j, & \text{if } \exists k > 0, \\ w(e''_{N+1}, t_m), & \text{otherwise;} \end{cases}$$

$$Q(t_{m+1}) = \begin{cases} DEQ(Q(t_m), k), & \text{if } \exists k > 0, \\ Q(t_m), & \text{otherwise;} \end{cases}$$

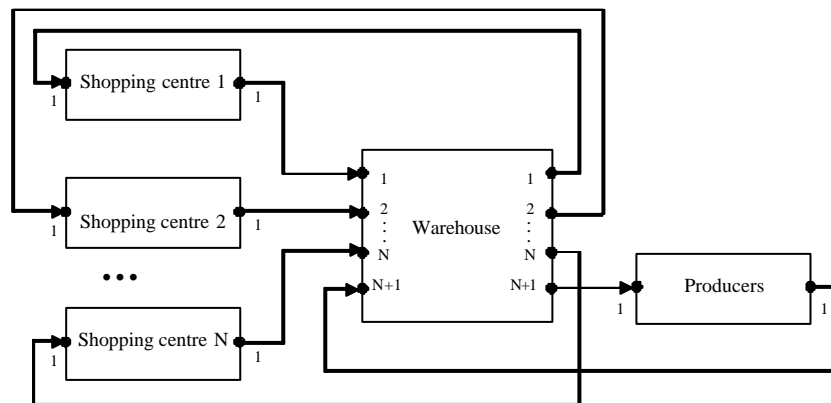
$$s(t_{m+1}) = \begin{cases} s(t_m) - n_2, & \text{if } \exists k > 0, \\ s(t_m), & \text{otherwise.} \end{cases}$$

$G(e''_{N+1})$:

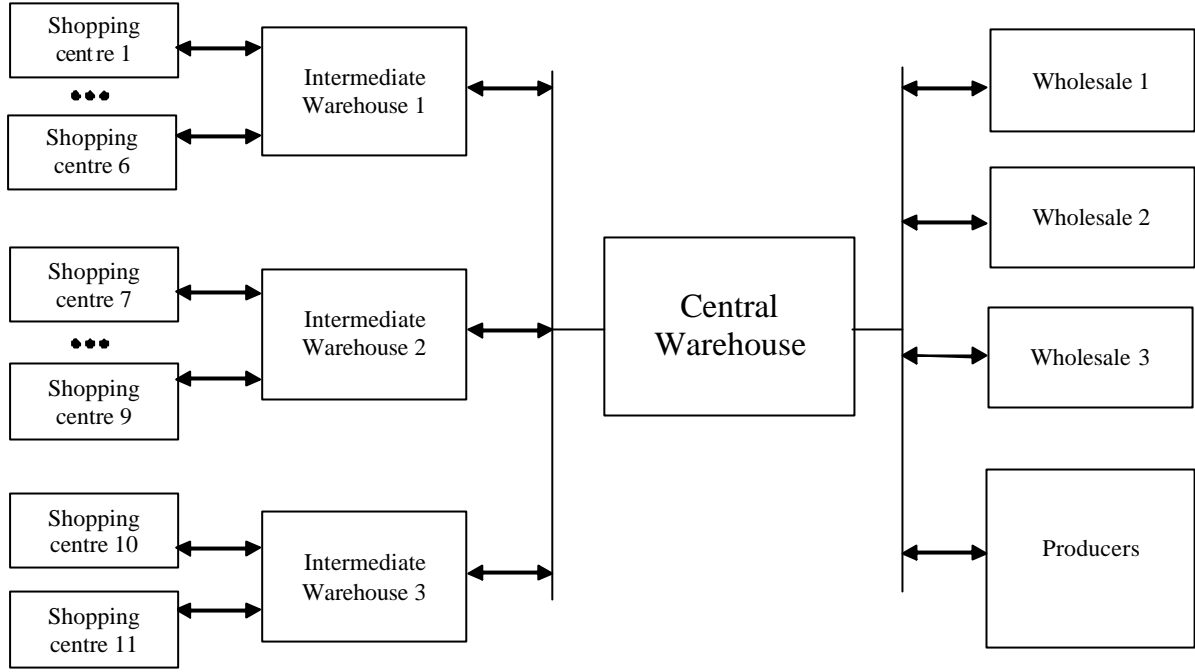
$$y_{N+1} = \Delta s, \text{ if } s(t_m) \geq s_0 > s(t_{m+1}).$$

Aggregate Producers

1. The set of input signals. $X = \{x_1\}$, where x_1 - request for delivery of goods.
2. The set of output signals. $Y = \{y_1\}$, where y_1 - amount of delivered goods.
3. The set of external events. $E' = \{e'_1\}$, where e'_1 - arrival of x_1 signal.
4. The set of external events. $E'' = \{e''_1\}$, where e''_1 - duration of time interval of goods delivering from Producers.
5. Controlling sequences. $e''_1 \mapsto \{\mathbf{x}_j\}$, $j = \overline{1, \infty}$.
6. Parameters. $z_n(t_m) = \{w(e''_1, t_m)\}$.
7. Discrete component of state. $\mathbf{n}(t_m) = \{d(t_m)\}$, where $d(t_m)$ - an amount goods delivered to Warehouse.
8. Initial state. $w(e''_1, t_0) = \infty$, $d(t_0) = 0$.



Figures 1: Aggregate scheme of specification



Figures 2: Aggregate scheme of simulated logistic system

9. *Transition and output operators:*

$H(e'_1)$:

$$w(e''_1, t_{m+1}) = t_m + x_j$$

$$d(t_{m+1}) = x_1.$$

$H(e'_1)$:

$$w(e''_1, t_{m+1}) = \infty, d(t_{m+1}) = 0.$$

$G(e''_1)$:

$$y_1 = d(t_m).$$

SIMULATION RESULTS

Formal specification presented in the previous section was used developing simulation model of commerce logistics centre. Simulation model was created using system PRANAS-2 (Pranevicius et al. 1994).

Integrated efficiency of logistic center is evaluated by the loss function $S = \frac{k}{T} \sum_i QS_i \cdot \Delta t_i + \frac{l}{T} \sum_i QW_i \cdot t_i$ where k - the coefficient which evaluated the loss of one unit of goods in LTL; l - the coefficient which evaluates the loss of storage of one unit of goods in LTL; QS_i - quantity of shortage of goods during time interval t_i ; QW_i - quantity of goods which are in shops or warehouses during time interval t_i .

The aggregate structure of simulated logistic system is presented in Fig.2. There are 3 intermediate warehouses, 1 central warehouse, 3 wholesales and 11 shopping centers. Time duration of simulation 2 years, unit of order is packet of goods which size is 270 units of goods ($l = 1.087$, $k = 10.8$)

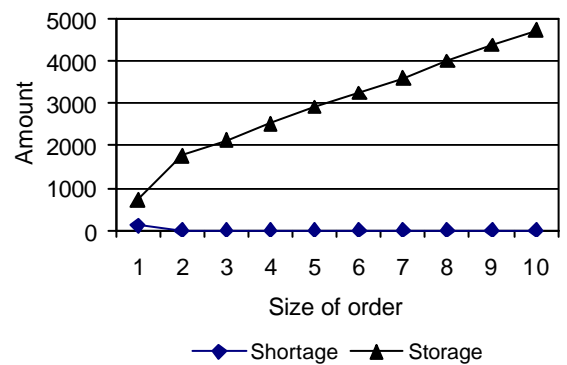
Other parameters of model:

Demand of goods in shopping centers are accidental and uniform distributed in interval $[0.3, 0.7]$ packets.

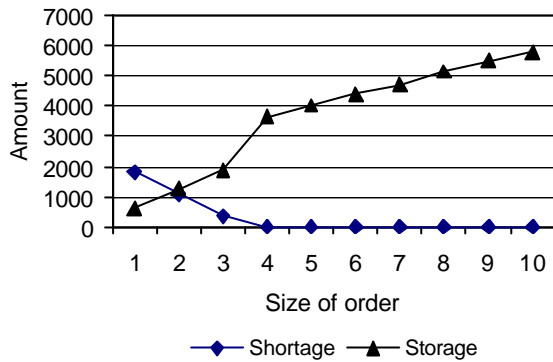
Demand of goods in wholesales are accidental and uniform distributed in interval $[3, 4]$ packets.

Initial amount of goods in shopping center are 3 packets. Initial amount of goods in wholesales are 5 packets. Goods delivery time to shopping center are 3 days. Initial amount of goods in intermediate warehouse are 8 packets. Orders processing time in intermediate warehouses are uniform distributed in interval $[1.4, 1.5]$ hours. Initial amount of goods in central warehouse are 40 packets. Orders processing time in central warehouse are uniform distributed in interval $[1.5, 2]$ hours. Goods delivery times from producers are uniform distributed in interval $[1, 3]$ days.

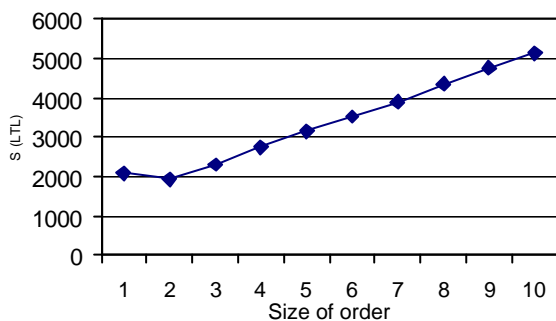
Simulation results are presented in Fig 3–6.



Figures 3: Shortage and storage functions for shopping centre



Figures 4: Shortage and storage functions for wholesale



Figures 5: Loss functions for shopping centre



Figures 6: Loss functions for warehouse

CONCLUSIONS

Carried out investigation showed that aggregate formalization approach can be successfully used developing simulation models of logistics systems. The main advantage of this approach is that it permits to analyze correctness of created formal specifications before generating simulation model.

REFERENCES

Bruzzone, A., Mosca, R., Briano, C., Brandolini, M. 2000. *Models For Supporting Customer Satisfaction in Retail Chains*. In: Proc. of International Workshop on Harbour, Maritime &

- Multimodal Logistics Modelling and Simulation. Portofino-Italy, p.161-164.
- Bruzzone, A.G., Mosca, R. 1999. *Modelling & Simulation and ERP Systems for Supporting Logistics in Retail*. In: Proc. of ESS 1999, Erlangen, Germany.
- Buslenko, N. 1971. "On a class of complex systems". Problems of applied mathematics and mechanics. Nauka, Moscow, pp56-68.
- Daganzo, F. 1999. "Logistic Systems Analysis." Springer-Verlag, Berlin.
- Lynch, N.A. 1996. "Distributed algorithms." Morgan Kaufmann Publisher, San Francisco.
- Pranevicius, H. 1991. *Aggregate approach for specification, validation, simulation and implementation of computer network protocols*. In: Lecture Notes in Computer Science, No 502, Springer-Verlag, p. 433-477.
- Pranevicius, H. 2000. *Formal Specification of Simulation Models Using Aggregate and Z Formal Methods*. In: Proc. of 2nd International conference "Simulation, Gaming, Training and Business Process Reengineering in Operations", September 8-9, Riga Latvia, p.350-354.
- Pranevicius, H., Pilkauskas, V., Chmieliauskas, A. 1994. "Aggregate approach for specification and analysis of computer network protocols." Kaunas, Technologija.
- Verbaeck, A., Saanen, Y.A., Valentin, E.C. 2000. *Logistic Modeling and Simulation for Automated Guided Vehicles*, In: Proc. of European Simulation Symposium and Exhibition, p. 514-519.

AUTHOR BIOGRAPHY

Professor, habil.dr. Henrikas Pranevicius is the head of the Business Informatics Department at the Kaunas University of Technology and the research group "Formal Specification, Verification and Simulation of Distributed Systems." The research group has been working in the field of creation of formal description methods for systems with distributed information processing and for modelling of complex systems for a long time. The results of investigations have been successfully applied creating the computerised system for specification, validation, and simulation/modelling of computer network protocols. H.Pranevicius has published over 200 scientific papers and 4 monographs.

SIMULATION METHOD OF THE PUBLIC LOGISTICS CENTER LOCALIZATION IN POZNAN CITY

Eugenia Narbuntowicz
Waldemar Walerjańczyk
Instytut Maszyn Roboczych i Pojazdów Samochodowych
Politechnika Poznańska
ul. Piotrowo 3
60-965 Poznań
Poland
E-mail: ewan@sol.put.poznan.pl

KEYWORDS

Facilities location, traffic, logistics center, transport cost

ABSTRACT

An argumentation to constitute The Logistics Centrum in Poznań has been discussed in this paper in a very wide range. The social, economic and ecology profits, which could give results from this fact, are obvious. This work presents mathematical models and methods, which have been selected to find optimum location for a facility. These methods determine where The Logistics Centrum should be located in Poznań.

INTRODUCTION

The political and economical changes in last years of 20-th century have showed that in Poland similar as in the other countries of Europe the displacement of goods is now dominated by road transport. There is prognosis that -in spite- of permanent development of European transport organization it will be not possible to full fill all carry needs. The situations are going worse in big cities, where road traffic could make very often to block all possibility to move every vehicle. Also we have to add environmental problems emanating from road traffic, such as a noise, air pollution, and vibration. Since of that it is indispensable to prepare new technological, economical and organization ideas together with methods of positive influences to organize zone, country and international transport systems. It seems that one of this idea is just the concept to establish Logistics Centruns t last.

The Logistics Centrum could be the specific economical and organizational complex, which has to stimulate to move of goods in the area, where it functions. Such Centrum makes all services not only in distributions of commodity from supplier (producer) to receiver (customer) but also makes many - in full or in some of range - in the matter of removing raw materials or trash. This complex form of services in connecting with its system organization, that is backed onto their own data bases and own network of representative agents gives predominance on the transport and store market.

Everything, which we said before, means that such Centrum gives possibility to coordinate moving materials, goods, and raw to consumers with minimum costs. What is more, they are (Centruns) able to give an application very modern methods in the transport of goods in regional, country and international areas using many kinds of transport, of course, with preferential to road transport.

It is worth also to take into account that the Centrum may be open for a cooperation mainly together with a network of independent business enterprises on the base of two-sided contracts and it can function in some time or permanently by making own representatives in those services, when there is no contracting partners, who will guarantee high standards and proof the function of all delivery system.

The choice of Poznań as a location for the future Logistics Centrum is necessary because of intensive economical development not only the city but also the whole region of Wielkopolska. This subject have been discussed for many years and because of that it would be possible to make better not only for moving goods but also labor, information and cash into this agglomeration. Here, in this town, would be done decisions according managing transport as well as transport peoples and goods in this area. The table below could to explain of passenger cars increase in the town.

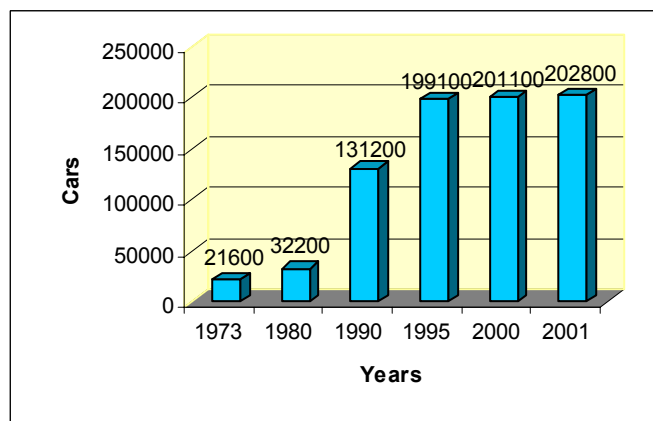


Fig.1 Increase of passenger cars in Poznań

The City of Poznań is the centre of attraction with the great economy potential, which gives enormous chance to realize this plan with full success. The last public probes show that

it is more attractive place (in spite of Warszawa) to invest then Kraków and Wrocław for example.

Tab.1 Subjects of economies in chosen cities

Statistical data	Poznań	Gdańsk	Kraków	Łódź	Warszawa	Wrocław
Area in km ²	261	262	327	294	494	293
Population in thousand	572,0	455,5	740,7	786,5	1609,8	634,0
Subjects of economies in thousand	81,0	56,9	94,7	86,3	244,8	90,6
In it Company of law merchant	7,9	6,1	7,6	5,1	37,8	6,9
In it With participation of foreign capital	2,0	1,1	1,5	1,2	13,0	2,0
Average employment in thousand .	151,9	77,0	164,0	121,0	775,3	117,2
Passenger cars in thousand .	202,8	161,2	255,0	217,8	-	219,1
Cars /1000 person	355	354	344	277	-	345
Places in cinemas	12845	5185	7372	5223	26722	4948
Hospital beds/10 thousand person	96,1	76,0	79,3	74,8	68,0	85,6
Places in hotel	3870	2001	6878	2228	14723	4257
Person using with hotel in thousands	389,4	313,7	792,2	191,1	1003,6	567,6
In it foreign tourists	138,0	125,6	400,2	37,1	585,4	143,7

In Poznań there is much kind of economic enterprises in such branches like: motorization, furniture, agriculture, machine building, food and chemical industry, production of cleaner and pharmaceutical media. We have also strongly developed finance market, retail and wholesale trade as well as tourist business.

The Logistics Centrum, if it would be organized in Poznań, will have influence onto local traffic flows that are connected with economical activity. Logistical operators in logistical centurms, who offers services, are concentrated onto costs minimize, among others, trough an integration of activities for many customers. It gives optimization of routs, choice of vehicles sizes for charges, integration of deliveries for several customers etc.. In result of these we can get not only reduction of costs but also decreasing a total amount of traffic and all negative influence that is make in the town.

LOCATION MODELS OF LOGISTICS CENTRUM

The Logistics Centrum (facility) location problem is one of the first and moat important element to make an investment and all enterprises will be affected. The next step requires to find some finance sources that are necessary for future investors to put in order all law and ground questions which are most suitable. Location models are based on an assumption of equilibrium, for which facility will attain minimum costs or maximum of profits. Most of these models are static and assume static equilibrium.

Mathematical form of a facility location is as following: let consider network, where facilities are described by vector $Y=(y_1,y_2,...,y_n)$ and they placed on the tops of this network Values of Y describes products set which are flowing across net in time unit. In a general way facilities location approves

some requirements, which are forced by vector $X=(x_1,x_2,...,x_m)$. Value of X describes quantitative requirements, which are demanded in "claim points" in particular time unit. On the other hand $K(X)$, $K(Y)$ there are data, which describe customers and facilities location. In other words, the location problem depends on fulfilling requirements and demands and, generally speaking, we can record it in the form as:

$$K(Y)=f(X,K(X))$$

There are many methods of facility location solution but we will present only two of them. In the first one the point of the matter is to divide an area into several zones and in relation to it determine optimum location of Logistics Centrum. Assume that we divide area into N zones with, more or less number of inhabitants (and/or economic enterprises), in order that this location of The Logistics Centrum minimize costs of transport. Let assume that L_i is the number of inhabitants or users in the zone and concentration of said inhabitants is in the point with co-ordinations (x_i, y_i) . Very often this point it is "centroid" of this zone i but not evermore is quite so. We also assume that ever of inhabitants or users bear the some cost of transport (we think about cost of fuel) b_i for one kilometer as well as that travel time in the zone is the some for all so it amount is v_i km/h and cost of "hour" for inhabitant is g_i . Then travel time for one kilometer is:

$$\frac{g_i}{v_i}$$

Then total cost per person – kilometer it is the sum time of driving per kilometer plus car cost per kilometer or:

$$c_i = b_i + \frac{g_i}{v_i}$$

It is obvious that we, here, had assumed some simplification, namely, that all costs are the same for all zones and in this case optimal location (which is understood as transport costs minimize) is depended from costs of transport.

Now we assume that The Centrum location has co-ordinations (x_f, y_f) . Let $d(f, i)$ will be a distance between zone i and The Centrum. In this case hole cost of transport for an inhabitant to get from zone to The Centrum is:

$$P_i \times d(f, i) \times c_i$$

and for us is to find a solution of problem how to make choice of The Centrum Location as minimum of costs sum through all zones i . it means to find minimum of a sum as below:

$$S = \sum_{i=1}^N L_i d(f, i) c_i$$

The distance we can deal with in two ways. First of all as a normal linear distance:

$$d_E(f, i) = \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2}$$

or using an absolute value

$$d_O(f, i) = |x_f - x_i| + |y_f - y_i|$$

but in this situation there is a problem since absolute value is not continuous function thus we would have to insert additional conditions into software. Taking into account that a distance is an positive value and that sequence of co-ordinates is not a matter of fact we square all elements and next make extraction of roots using this form of measure for computation.

$$d_O(f, i) = \sqrt{(x_f - x_i)^2} + \sqrt{(y_f - y_i)^2}$$

Another method to estimate a location of The Logistics Centrum is a center of gravity model. In the more simple model the point of matter is to calculate the weighted average of customers locations. The point we will obtain would represent the best location for The Logistics Centrum. The counting model is as following:

$$X_o = \frac{\sum X_j D_j}{\sum D_j}; \quad Y_o = \frac{\sum Y_j D_j}{\sum D_j}$$

where:

D_j - demands of customer j ,

X_j, Y_j -customer j location co-ordinates,

X_o, Y_o -co-ordinates of the center of gravity.

The above model takes into account only locality of customers. If we want to take into account also e.g. time or cost of travel then our model will look as follows:

$$X_o = \frac{\sum T_j X_j D_j}{\sum D_j}; \quad Y_o = \frac{\sum T_j Y_j D_j}{\sum D_j}$$

where:

D_j - demands of customer j ,

T_j - costs of transport for link j ,

X_j, Y_j - co-ordinates of customer j location,

X_o, Y_o - co-ordinates of the center of gravity.

The next way to find optimal location for The Logistics Centrum is using an iterative method. It is heuristics procedure in its nature. We begin from a single location and by using iterative methods we come to the moment, where $X_o(t+1) - X_o(t)$ and $Y_o(t+1) - Y_o(t)$ are the (small) values, which will give us satisfaction. The counting model now is as follows:

$$X_o(t+1) = \frac{\sum \frac{m_j(t)}{d_j(t)} X_j}{\sum \frac{m_j(t)}{d_j(t)}}; \quad Y_o(t+1) = \frac{\sum \frac{m_j(t)}{d_j(t)} Y_j}{\sum \frac{m_j(t)}{d_j(t)}}$$

where:

$m_j(t)$ -weight factor for the link of customer j with The

Centrum, which is located in $(X_o(t), Y_o(t))$ point,

$d_j(t)$ - distance from The Centrum located in $(X_o(t), Y_o(t))$

point from customer j ,

X_j, Y_j - co-ordinates of customer j .

The methods, which we have presented above, are suitable only for single location and comparatively are simple procedures. The problem becomes much more difficult, when we have to find more locations or we look for new locations taking into account existing ones. So, we have to say that the standard linear methods are not sufficient to use for more complex procedures of locations finding. In this case then very often is used model as follows:

$$\sum_{ij} C_{ij} X_{ij} + \sum_i F_i Y_i \rightarrow \min$$

$$Y_i \in \{0,1\}, \sum X_{ij} = D_j, \\ 1 \leq i \leq n; \quad 1 \leq j \leq m$$

where:

n – number of facility,

m – number of customers,

F_i – cost with i facility,

X_{ij} – demand j customer with i facility,

C_{ij} – transport cost from customer i to facility i,

D_j – total demand customer j,

Y_i – value 1,0 (open/close facility i).

THE COMPUTER SIMULATION

The Logistics Centrum location problem in Poznań involves choosing the best place from the possible sets. The computer simulation has been done with real data by help of TransCAD computer system using Routing and Logistics (Facility Location) module and vector map of Poznań. The tables with addresses of potential customers have been introduced to the program. These data have been associated with the map of Poznań throughout addresses. The goal of this method is *minimum* average cost of service, so that all matrixes, which are necessary (distances, costs and time transport, service) have been counted. The computation, using data prepared in this way has been used to the next step. In result, we have got the place of The Logistics Centrum location (Fig.2).



Fig. 2 Map of Poznan with potential customers

Location algorithm model in TransCAD system works in two steps:

- it identifies a set of initial facility locations using a greedy heuristic,
- it attempts to improve on the initial set of locations by swapping candidates with chosen facilities on a pair-wise basis until no improvement can be made.

The greedy heuristic chooses the next best location by evaluating all the candidates and selecting the one best to achieve the desired objective.

Unfortunately, the location that we have found, as the result of above activities, is placed just in the exact center of the

town and because of that is not possible to be taken into account as a optimum location for The Logistics Centrum (among others for the sake of high prices of ground) and therefore we have used different method of investigation.



Fig. 3 Candidates and the best location of Logistics Center in Poznań

The point of the matter in this method is to before select candidate locations, which will full fill conditions for The Logistics Centrum location in a future. We have selected five such locations Fig. 3 taking into account first of all sizes and prices of grounds. The TransCAD computer program procedures has been used to further elimination, when we have introduced to it data for these five new locations. Then we have counted again matrixes: distances, costs and time of transport and service for these widen data of logistics centurms locations(customers remain the same) as well as we have introduced weights for any customer and centurms. In the result of computing we have got answer, which of the proposed location for The Logistics Centrum is the best one. It has appeared that this location which full fill all requirements of optimal location and is placed at Walecka Street in the north-west part of Poznań.

REFERENCES

- P.B. Mirchardani and R.L. Francis 1990. "Discrete Location Theory". John Wiley & Sons, Inc., New York.
- F. Rosing, 1992. "An optimal method for solving the (generalized) multi-Weber problem". European Journal of Operational Research, Vol. 58, p. 414 – 426
- E. Narbuntowicz, W. Osmólski, W. Walerjańczyk 1997, „Zarządzanie flotą pojazdów - systemy decyzyjne”. pp 361-369, Zweite Deutsch-Polnische Logistikkonferenz, Poznań.

SIMULATION IN MANUFACTURING

Manufacturing Performance Simulation

IMPROVEMENT OF THROUGHPUT AND RESOURCE UTILIZATION IN AN AUTOMOTIVE STAMPING PLANT

Edward J. Williams
Marcelo Zottolo
Production Modeling Corporation
Three Parklane Boulevard, Suite 1006 West
Dearborn, Michigan 48126
United States
{ewilliams mzottolo}@pmcorp.com

KEYWORDS

Industrial processes, Performance analysis, Discrete simulation, Simulation interfaces, Process-oriented.

ABSTRACT

Simulation has long been recognized as an analytical tool of high power and wide applicability when applied to the improvement of manufacturing processes. Indeed, historically, manufacturing applications were the first major purview of simulation usage by large companies. As manufacturing processes increase in complexity, both operational and economic, the capabilities of simulation increase both in importance and in difficulty of duplication by alternative analytical methods. In this paper, we examine in detail the application of simulation to an automotive stamping plant. Simulation identified bottlenecks in material handling, pointed the way to increasing utilization of a costly stamping press, and quantified the relationships between specific capital investments under consideration and the throughput increase to be expected.

INTRODUCTION

Manufacturing industries and their complex processes have long been justifiably recognized as the earliest and still among the most frequent users of simulation for productivity improvement (Miller and Pegden 2000). In turn, the automotive industry, among the largest participants within the manufacturing sector in many countries (the United States, in this case) and hence a bellwether of the national economy, has traditionally maintained and enhanced its competitiveness with the help of simulation analyses to increase efficiency within its extensive manufacturing operations (Ülgen and Gunal 1998).

Here, we examine a case study of simulation application to productivity improvement within one stamping plant owned and operated by a major automotive company in the United States. This client wished to increase throughput in response to changing and increasing market demand while simultaneously increasing utilization of the expensive

stamping press (indeed, the keystone of the process) and reducing congestion among the material-handling vehicles responsible for delivering panels to, and taking them away from, this stamping press. After providing an overview of the manufacturing process, we describe the construction, verification, and validation of the simulation model used in the analysis of this process. We then outline the results and recommendations obtained from the model, and indicate the directions of expected follow-up analyses. Numerous precedents of this type of study appear in the literature. For example, (Hoff et al. 1997) focused attention on automated guided vehicle systems; (Mosca, Queirolo, and Tonelli 2002) focused on job-sequencing in a system, like this one, dominated by an expensive machine; and (Mossa and Mummolo 2002) examined a manufacturing system characterized by intermixed automatic and manual operations.

THE MANUFACTURING PROCESS AND ITS CONCERNS

Since the press is the centerpiece of the production operations within the scope of the study, the manufacturing process is conveniently viewed and described as partitioned into one portion upstream from the press (input side) and one downstream (output side) from the press. On the upstream side, an SGV [self-guided vehicle] (Heizer and Render 2004) fetches a pallet, loaded with many (potentially more than 200; the specific quantity a system variable) steel blankets, from the input supply dock. This pallet is transferred first to the loaded pallets stand and then, via forklift and operator, to the empty destacker. After the forklift operator removes all dunnage from the steel blankets, the loaded pallet is shuttled to the press, which empties it by withdrawing one blanket at a time for the stamping operation. Subsequently, the empty pallet is shuttled out of the press via the destacker and transferred, again via forklift and operator, to the empty pallet stand and thence to an SGV. The SGV returns the empty pallet to the dock and returns with a full pallet, thereby completing the upstream operational cycle of supply to the press. These SGVs are powered by lead-acid storage batteries, which must periodically be recharged; to operate safely and effectively, an SGV must have between 20% and 80% of

the theoretically maximum charge. Therefore, each SGV must be monitored and routed to one of the charging spurs in the system when its charge level approaches the lower threshold. An SGV exhausts its charge much more rapidly when moving than when standing still. The complexities introduced into the system and its model representation by these considerations resemble the complexities examined for the same reason in a study by (Norman 2002).

On the downstream side, the press feeds three vertical conveyance systems [VCSs] via two conveyor belts, as depicted in Figure 1.

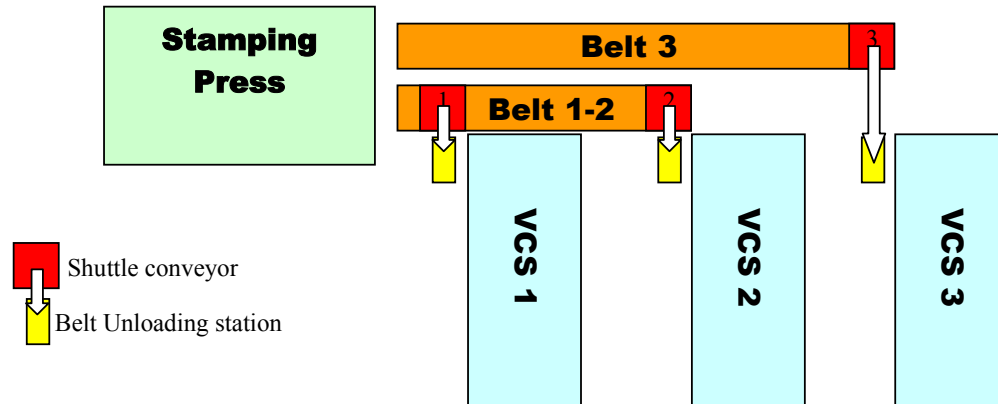


Figure 1: Downstream Material Handling of Press Output

Downtimes, when they occur, significantly affect material routing. If shuttle conveyor #1 is down (worst case), neither VCS 1 nor VCS 2 can be fed; the press can send output only to VCS 3 via belt 3. If shuttle conveyor #2 is down, VCS 2 cannot be fed; the press will alternate its output between VCS 1 and VCS 3. If shuttle conveyor #3 is down, the press will alternate its output between VCS 1 and VCS 2 via belt 1-2. No parts can be routed to VCS n ($n = 1, 2$, or 3) if it is down, since accumulation of parts in either belt is forbidden.

CONSTRUCTION, VERIFICATION, AND VALIDATION OF THE MODEL

At the client's request, the model was developed using the WITNESS® simulation software tool marketed by the Lanner Group worldwide. This software provides its own internal programming-logic language; simulation constructs such as machines, buffers, vehicles, and tracks; concurrent development of a simulation model and its accompanying two-dimensional animation, nearly automatic provision for basic statistical output reports, and ability to read input data from Microsoft® Excel workbooks (Mehta and Rawles 1999). The client had previously adopted the policy of developing almost all simulation models using one of only two software tools (one of these being WITNESS®) to achieve better return on the investment of training its own industrial, production, and/or process engineers in simulation and to make possible easier transferability of models, among these engineers and consultant engineers collectively, in different corporate departments or divisions (Williams 1996).

To encourage frequent use of the model among client engineers (not all of whom know the details of WITNESS® software) to explore various alternatives, and to make such use more convenient, more rapid, and less prone to input data error, system variables likely to change were placed into worksheets within a Microsoft® Excel workbook – as vigorously recommended by many expert practitioners of simulation (Ülgen et al. 1994) – and thence read into the model at initialization time. Examples of variables thus input were the numbers of SGVs at the input and output sides of the press, the speeds of these vehicles either loaded or unloaded, the number of blanks per pallet, the press rate,

and numerous pallet transfer times within the system.

The client and consultant engineers agreed upon and documented underlying model assumptions, or restrictions of its scope, necessitated by the confluence of absence of data and project time constraints. The time constraints were imposed by client management due to exigencies of making capital-expenditure decisions and implementing production increases and improvement efficiencies on short notice. Examples of these accommodations to a tight schedule included:

1. All times to fail and times to repair are exponentially distributed.
2. The model will run only one part type at a time, without inclusion or examination of changeover times.
3. Production runs on three abutting eight-hour shifts per business day.
4. Operators and forklift trucks have zero downtime.

Here too, the explicit acknowledgment and documentation of assumptions, forestalling potential later misapplication of a simulation model within a context it is unequipped to handle, represent practices vigorously expounded on the basis of experience (Musselman 1994).

Construction, verification, and validation of the model were naturally divided into two major components – the upstream side of the press, and the downstream side of the press, matching the canonical view of the manufacturing process as described in the previous section. The press itself was represented as a WITNESS® “machine;” the blankets it stamped were represented as WITNESS® “parts.” The SGVs responsible for material handling and

transport were represented as WITNESS® “vehicles” running on WITNESS® “tracks.” In WITNESS® modeling methodology, a vehicle boards a track at the rear and travels to the front, and the logic specifying where (i.e., to which other track) the vehicle should go next is specified within the track, not within the vehicle. Whenever a vehicle reached the front of the track, logic checked the level of its battery charge; if this level was too low to allow the vehicle to proceed on its canonical route, it was routed to a charging spur. On such a charging spur, the vehicle spent the required recharging time loading a dummy “part” conceptually representing electrical energy. Operators were represented as WITNESS® “machines.”

Verification of the model entailed numerous walkthroughs, in which a total of four analysts participated, especially of the complex internal code specifying SGV routing logic. Additional verification techniques used included desk checking, traces, and careful examination of the animation accompanying the simulation model (Carson 2002). Validation techniques used in several meetings among the client and consultant engineers included walkthroughs of the model entities, examination of the animation, and Turing tests (Law and Kelton 2000).

MODEL RESULTS AND OUTPUT

First, the model was run with no equipment downtime. Equipment downtime can cause the parts material flow to stop. The stamping press might be forced to slow down

Press starvation (idleness) represented how robustly the press input side material handling system is designed. In the same way, press blockage represented how robustly the press output side material handling system is designed.

- When only one forklift truck is available at the Market Place (the final downstream destination, beyond Figure 1, for stamping press output), increasing the number of SGVs available (at the output side of the press) does not reduce the press blockage (about 60%). This is true for VCS operator’s cycle time of 11, 15 or 20 seconds. It is evident that, when only one forklift truck is available at the Market Place and under the given assumption of 2 minutes to load or unload an SGV cart, the number of SGVs used at the output side and the operator’s cycle time at the VCS have no impact on the press performance statistics.
- The same effect discussed in the previous item is observed when only two forklift trucks are available at the Market Place. Under the assumption of 2 minutes to load OR unload an SGV cart and 3 carts per SGV, regardless of the operator’s cycle time, increasing the number of SGVs does not reduce the press percentage blockage. Increasing the number of SGVs at the output side only causes the queue for the Market Place to increase. SGVs cannot leave the Market Place any oftener than 1 every 6 minutes.
- For any combination of number of SGVs at the input and output side and VCS operator’s cycle times of 11

System Parameters			
<i>Number of SGVs on the Press INPUT side</i>	4	5	
<i>Number of SGVs on the Press OUTPUT side</i>	10	12	14
<i>Number of initial trays per MMI system</i>	8	10	
<i>Operator's cycle time to load a part in a rack</i>	11	15	20
<i>Number of forklift trucks (parallel load/unload stations) at the Market Place</i>	1	2	3

Table 1: Parameters and Values Examined During Experimentation

(i.e., suffer blockage) depending on which equipment fails (location in the material handling chain) and the duration of the failure. Leaving the downtime aspect out of the experimentation allowed identifying any and all blockages of the press due to material handling capacity.

Several operating conditions were evaluated, as indicated in the table below:

For the experimentation, various combinations of all of these parameters were utilized. Thirty-six scenarios were run, each with a warm-up of 24 hours and a run time of 5 days.

As previously stated, the stamping press operating statistics were the main performance metric under investigation.

or 15 seconds, as the number of forklifts (loading/unloading stations) at the Market Place is increased, press blockage is reduced. Increasing the number of SGVs at the output side helps reduce press blockage by allowing a more efficient delivery of stamped parts to the Market Place. However, when the operator’s cycle time is 20 seconds, making available a third forklift truck does not help reduce the percentage block of the press. It seems that for operator’s cycle time of 15 seconds and above, increasing the number of forklift trucks stops having a positive impact on the press performance. Therefore, depending on the range of the operator’s cycle time utilized, the number of forklift trucks at the Market Place might have a significant positive impact on the press performance or no impact at all.

- 8 to 10 trays are adequate for maximum VCS efficiency.
- At the assumed press rate (750 strokes/hour – worst case scenario), the operators manually loading stamped parts at the VCSs receive a set of parts from the press every 14.4 seconds. When three forklift trucks are available at the Market Place, assigning a loading cycle of 11 seconds allows this manual operation to keep up with the press rate. Under this Market Place and operator's cycle time conditions, 10 SGVs at the output side are enough to avoid blocking the press. However, when only 4 SGVs at the input side are allocated and under the previously mentioned conditions (11-second cycle time, 10 SGVs, and 3 forklifts), they cannot continuously feed the press and approximately 1.3% press starvation is observed. With 5 SGVs, no starvation appears. This statement holds true under any of the 36 scenarios evaluated. Therefore, minimum SGVs required are 5 and 10 at the upstream and downstream side of the stamping press, respectively.
- Due to different downtimes (stamping press, distribution conveyors, VCSs, etc.) the average throughput is 671 strokes/hour when the press runs at 750 strokes/hour.
- At a press rate of 750 strokes/hour, one part every 4.8 sec is stamped. Considering a rack capacity of 20 pieces, it takes 96 seconds to fill a rack. SGVs transport three trays (with one rack in each), therefore, an SGV is needed at the VCSs every 4.8 minutes ($4.8 \times 20 \times 3/60$). As a consequence, SGVs should be released from the Market Place every 4.8 minutes to stay abreast of the stamping press rate. The chart below shows how press performance is affected by the time it takes to service an SGV (unload it and reload it)

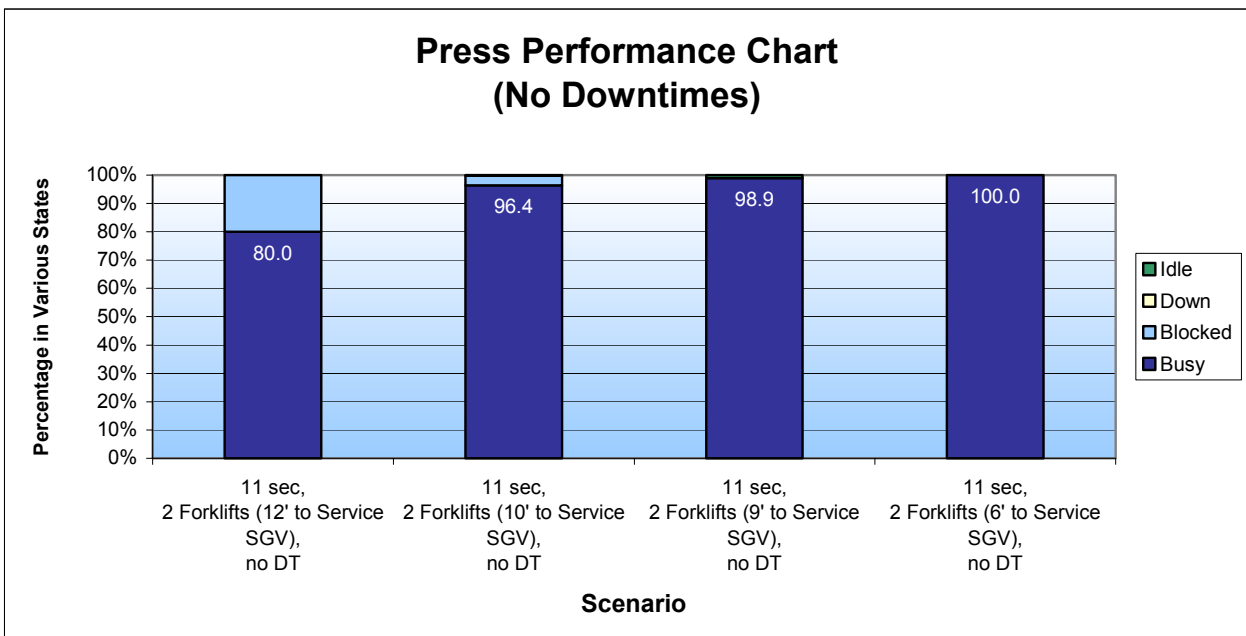
minutes to service an SGV at the input dock. The fourth bar assumes 4 minutes to service an SGV at the input dock.

- Simulation analysis improved locations for SGV opportunity chargers in front side of the press.

CONCLUSIONS AND FUTURE WORK

The simulation study described here has provided significant help and value to the client by confirming the throughput of the system under a variety of operating conditions, identifying proportions of time capital-intensive resources (e.g., machines, material-handling equipment, and operators) spend in states of starvation, busy, blocked, idle, or down, and by assessing the sensitivity of the upstream side of the system to the number and location of recharging spurs available to the SGVs used there. In meetings held with the client engineers and managers as the current phase of the study drew to a close, the simulation analysts presented extensive written documentation, both internal and external to the model itself, to assist all parties in implementing planned extensions to the model; the importance of such documentation, relative to achieving both model credibility and model usability over a lifetime equal to that of the process under study, has been extensively discussed in the literature, particularly in the seminal paper (Oscarsson and Moris 2002).

These planned extensions to this study currently include implementing the ability to run multiple part types, thus entailing incorporation of die changeover times into the model, incorporation of operator break times within the model, and assessing the sensitivity of the system to a third belt ("belt 4" if the nomenclature above is extrapolated) on the downstream side of the system. That belt currently exists, but is used only as a backup material-handling device.



at the Market Place. Below 9 minutes it is no longer a system constraint. The first three bars assume 8

ACKNOWLEDGMENTS

The authors extend warm appreciation to Dr. Onur M. Ülgen, president of Production Modeling Corporation and senior professor at the University of Michigan – Dearborn, for his mentoring, support, and guidance throughout this project. Additionally, an anonymous reviewer provided valuable suggestions to improve this paper.

REFERENCES

- Carson, John S., II. 2002. "Model Verification and Validation." In *Proceedings of the 2002 Winter Simulation Conference*, Volume 1, eds. Enver Yücesan, Chun-Hung Chen, Jane L. Snowdon, and John M. Charnes, 52-58.
- Heizer, Jay, and Barry Render. 2004. *Operations Management*, 7th edition. Upper Saddle River, New Jersey: Pearson Education, Incorporated.
- Hoff, Axel, Holger Vogelsang, Uwe Brinkschulte, and Oliver Hammerschmidt. 1997. Simulation and Visualization of Automated Guided Vehicle Systems in a Real Production Environment. In *Proceedings of the 9th European Simulation Symposium*, eds. Winfried Haha and Axel Lehmann, 391-395.
- Law, Averill M., and W. David Kelton. 2000. *Simulation Modeling and Analysis*, 3rd edition. Boston, Massachusetts: The McGraw-Hill Companies, Incorporated.
- Mehta, Arvind, and Ian Rawles. 1999. "Business Solutions Using WITNESS." In *Proceedings of the 1999 Winter Simulation Conference*, eds. Phillip A. Farrington, Harriet Black Nembhard, David T. Sturrock, and Gerald W. Evans, 230-233.
- Miller, Scott, and Dennis Pegden. 2000. "Introduction to Manufacturing Simulation." In *Proceedings of the 2000 Winter Simulation Conference*, eds. Jeffrey A. Joines, Russell R. Barton, Keebom Kang, and Paul A. Fishwick, 63-66.
- Mosca, Roberto, Filippo Queirolo, and Flavio Tonelli. 2002. Job-Sequencing Problem in a Semi-Automated Production Process. In *Proceedings of the 14th European Simulation Symposium*, eds. Alexander Verbraeck and Wilfried Krug, 343-347.
- Mossa, Giorgio, and Giovanni Mummolo. 2002. Simulation Modelling of both Automated and Human-Based Manufacturing and Assembly Line. In *Proceedings of the International Workshops on Harbour, Maritime, & Multimodal Logistics Modelling and Simulation; Modelling & Applied Simulation 2002*, eds. Agostino G. Bruzzone, Yuri Merkurjev, and Roberto Mosca, 103-111.
- Musselman, Kenneth J. 1994. "Guidelines for Simulation Project Success." In *Proceedings of the 1994 Winter Simulation Conference*, eds. Jeffrey D. Tew, S. Manivannan, Deborah A. Sadowski, and Andrew F. Seila, 88-95.
- Norman, Michael. 2002. "A Recursive Method for Traffic Management Through a Complex Path Network." In *Proceedings of the 2002 Winter Simulation Conference*, Volume 1, eds. Enver Yücesan, Chun-Hung Chen, Jane L. Snowdon, and John M. Charnes, 537-541.
- Oscarsson, Jan, and Matías Urenda Moris. "Documentation of Discrete Event Simulation Models for Manufacturing System Life Cycle Simulation." In *Proceedings of the 2002 Winter Simulation Conference*, Volume 2, eds. Enver Yücesan, Chun-Hung Chen, Jane L. Snowdon, and John M. Charnes, 1073-1078.
- Ülgen, Onur M., John J. Black, Betty Johnsonbaugh, and Roger Klungle. 1994. "Simulation Methodology in Practice – Part II: Selling the Results." *International Journal of*

Industrial Engineering – Applications and Practice 1(2):129-137.

- Ülgen, Onur M., and Ali Gunal. 1998. "Simulation in the Automobile Industry." In *Handbook of Simulation*, ed. Jerry Banks, New York, New York: John Wiley & Sons, Incorporated, 547-570.
- Williams, Edward J. 1996. "Making Simulation a Corporate Norm." In *Proceedings of the 1996 Summer Computer Simulation Conference*, eds. V. Wayne Ingalls, Joseph Cynamon, and Annie V. Saylor, 627-632.

AUTHOR BIOGRAPHIES

EDWARD J. WILLIAMS holds bachelor's and master's degrees in mathematics (Michigan State University, 1967; University of Wisconsin, 1968). From 1969 to 1971, he did statistical programming and analysis of biomedical data at Walter Reed Army Hospital, Washington, D.C. He joined Ford Motor Company in 1972, where he worked until retirement in 2001 as a computer software analyst supporting statistical and simulation software. After retirement from Ford, he joined Production Modeling Corporation, Dearborn, Michigan, as a senior simulation analyst. Also, since 1980, he has taught evening classes at the University of Michigan, including both undergraduate and graduate simulation classes using GPSS/H™, SLAM II™, SIMAN™, ProModel®, or SIMUL8®. He is a member of the Institute of Industrial Engineers [IIE], the Society for Computer Simulation International [SCS], and the Michigan Simulation Users' Group [MSUG]. He serves on the editorial board of the *International Journal of Industrial Engineering – Applications and Practice*. During the last several years, he has given invited plenary addresses on simulation and statistics at conferences in Monterrey, México; İstanbul, Turkey; Genova, Italy; and Rīga, Latvia. His email and web addresses are <ewilliams@pmcorp.com> and <www-personal.umd.umich.edu/~williams>.

MARCELO ZOTTOLO, born in Buenos Aires, Argentina, came to the United States to finish his college studies. He was graduated from the University of Michigan - Dearborn as an Industrial and Systems Engineer in December 2000. He was awarded the Class Honors distinction and his Senior Design Project was nominated for the Senior Design Competition 2001. This project studied the improvement of manufacturing processes for the fabrication of automotive wire harnesses, ultimately proposing an automation tool leading to improvements in future designs. Additionally, he was co-author of a paper on simulation in a distribution system which earned a "best paper" award at the Harbour, Maritime, and Simulation Logistics conference held in Marseille, France, in 2001. He is currently working for Production Modeling Corporation as a Simulation Engineer. There, his responsibilities include building, verifying, validating, and analyzing simulation models in WITNESS and SIMUL8 for large corporate clients; he also presents in-house training seminars. His email address is <mzottolo@pmcorp.com>

MODULAR DESIGN AND SIMULATION BASED VERIFICATION OF THE LOGIC CONTROL CODE FOR AN AGILE SHOE MANUFACTURING SYSTEM

Emanuele Carpanzano and Andrea Cataldo
Institute of Industrial Technologies and Automation
National Research Council
Viale Lombardia 20/a, 20131, Milan, Italy
E-mail: e.carpanzano@itia.cnr.it, a.cataldo@itia.cnr.it

KEYWORDS

Manufacturing, Logic Control, Design Methodology, Discrete Simulation.

ABSTRACT

To properly deal with the increasing complexity of industrial distributed and agile manufacturing control systems, the use of methods and tools that support the designer in the definition, verification and implementation of logic control software is more and more necessary. Nowadays, many different concepts, methodologies, standards and computer aided design tools are available to face the considered problem in a structured and systematic way. As a consequence, critical decisions have to be undertaken to select the most appropriate techniques and tools to use, by considering their limits as concerns the provided functionalities, the compliancy with existing standards, and the openness towards not proprietary software instruments. In the present work a methodological approach to the structured design and simulation based verification of manufacturing plants logic control systems is discussed. The methods and tools used during the different phases of the design are illustrated with reference to an application example in the shoe manufacturing sector, with particular attention to the design and the closed loop discrete simulation of the PLC code for the control of the making department transportation line, that is based on an innovative molecular architecture.

INTRODUCTION

Reliable and flexible automation systems are a crucial point for competitiveness of modern manufacturing systems. Strong reusability and quick reconfigurability of developed control solutions are also fundamental elements to achieve agility, in order to reduce the costs and times needed to design and realize a new production systems, or to modify existent ones. In the present work a structured development methodology for the design of manufacturing logic control systems that achieve the mentioned objectives is applied to an innovative production system, and the adopted standards and tools are discussed. In detail, the methodology is based on the following main design phases (Ferrarini and Carpanzano 2002):

System definition: the process to be automated is described, and the activities to be performed and the objectives of the automation system are defined.

Control system specification: the tasks and the essential functions of the supervision and control system are defined, and the control strategy is established.

Control system design: the supervision and control systems are developed through suitable reference models for both the *architectural* and the *functional* design. Then, the control code is generated.

Control system verification: it is checked by means of discrete simulation based methods if the designed control functions meet the requirements (Carpanzano and Ballarino 2002).

System integration and testing: the control code is implemented on the target controllers, and it is checked if the control system works on its devices, and if all the system's requirements are met.

In the paper the different phases of the PLC (Programmable Logic Controller) code design and the used closed loop simulation based verification methods and tools are discussed with reference to the considered application example.

SYSTEM DEFINITION

In the considered agile shoe manufacturing plant (figure 1) an innovative transport line is used to move the semifinished shoes from a machining station to another one according to a predefined operation schedule.

The innovative molecular structure of the transport line of the making department (figure 2) enhances the modularity, scalability, integrability and reconfigurability properties of the production system, so increasing the overall flexibility of the plant.

The basic element of the molecular structure is the “Tern”, which is constituted by two rotating tables, called “Table” and “Island”, and by a rotating three arms manipulator.

The Table is used to direct the semifinished shoes either to the next Tern or to the Island of the same Tern. Moreover, it moves backward the lasts flowing back towards the warehouse (the last is the object around which the semifinished shoe is built upon).

The Island directs the semifinished shoes towards the different machining stations, laid around the Island itself.

The transportations of the semifinished shoes and lasts between Tables and Islands are carried out by the manipulators.



Figure 1: Integrated Pilot Plant

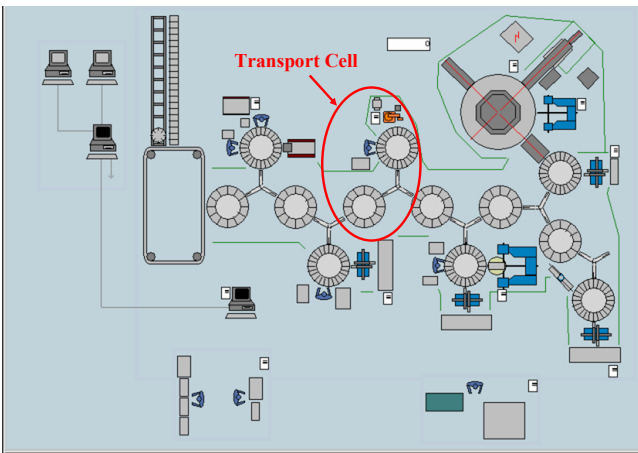


Figure 2: Integrated Pilot Plant Layout

CONTROL SYSTEM SPECIFICATION

To achieve the desired agility objectives, object-oriented concepts have been exploited for the control system development (Maffezzoni et al. 1999). In particular, the molecular line has been considered as a set of interacting Terns, each one with its own independent control system. So, each Tern operates autonomously according to its boundary conditions, while interfaces among different Terns are represented by semifinished shoes and lasts exchanges. With reference to a generic single Tern, the possible movements that involve a generic semifinished shoe or last are depicted by means of arrows in figure 3.

A semifinished shoe that arrives on the Table of a Tern (arrow H) and which has not to be machined by the Tern itself is directly moved to the next Tern (arrow C); otherwise, it is moved to the Island (arrow F), where it is properly machined (arrows D and E). For the sake of simplicity only one machining station has been here represented. As soon as the machining is over, the shoe is moved towards the next Tern (arrow G).

Whenever a shoe is finished, it is removed from its last, and the last itself goes back to the warehouse by flowing back through the whole transport system. So, for a generic Tern, the last that must be stored in the warehouse comes from the

adjacent Tern (arrow B) and goes back, through the Table, to the previous Tern (arrow A). Specifically, the lasts start their way back to the warehouse in the last Tern, once the shoe working process is over (arrow I).

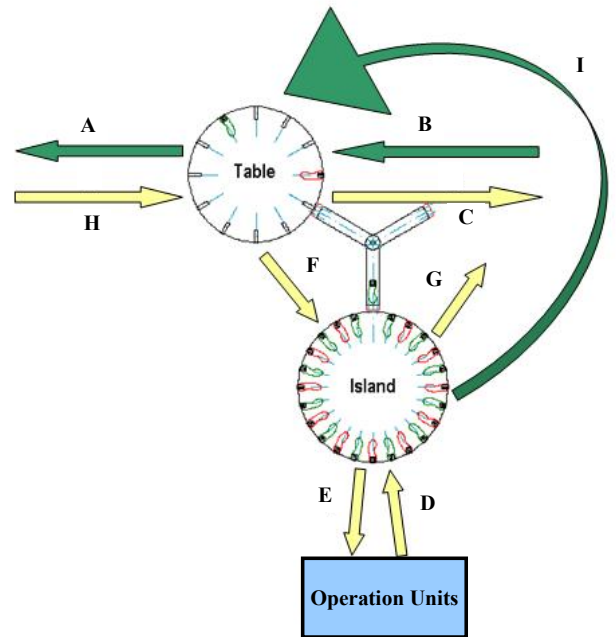


Figure 3: Tern Control System Specification

The proposed control strategy for a generic Tern is based on the following three basic assumptions:

1. avoid deadlocks, i.e. avoid situations into which none semifinished shoe or last can be moved;
2. favour the backward lasts flow respect to the forward one of the semifinished shoes;
3. favour the unloading of the Tern resources, i.e. of the Table, Island and Operation Stations.

Such heuristic rules have been defined to guarantee a correct and efficient use of the system resources.

To implement the first assertion it has been decided to have always one slot free on a Table for the last backward movement toward the warehouse. Moreover, to implement the second and third assertions proper priorities have been assigned to the different possible operations illustrated with reference to figure 3. Such priorities are represented in table 1. In the columns of the table the different operations have been listed. The presence of the character "x" in a cross between a column and a row means that the operation associated to the column has minor priority than the one associated to the row. Notice that the possibility to execute concurrent operations, e.g. operations A and H, is also considered in the table, e.g. operation AH. As a consequence, it is not significant to define a priority between operations A and H, since the concurrent operation AH can be carried out.

Table 1: Priority Logic for the Tern

	AH	A	BFG	BC	BF	BG	B	C	D	E	FG	F	G	H	IF	I
AH	-	X	X	X	X	X	X	X	-	-	X	X	-	X	X	X
A	-	-	X	X	X	X	X	X	-	-	X	X	-	N.O.	X	X
BFG	-	-	-	X	X	X	X	X	X	X	X	X	X	X	X	X
BC	-	-	-	-	X	X	X	X	-	-	X	X	X	X	X	X
BF	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	X
BG	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X
B	-	-	-	-	-	-	-	N.O.	-	-	X	N.O.	N.O.	X	X	X
C	-	-	-	-	-	-	-	-	-	-	X	X	X	X	-	-
D	-	-	-	-	-	-	-	-	-	X	X	X	X	-	-	-
E	-	-	-	-	-	-	-	-	-	-	X	X	X	-	-	-
FG	-	-	-	-	-	-	-	-	-	-	-	X	X	X	-	-
F	-	-	-	-	-	-	-	-	-	-	-	-	N.O.	X	-	N.O.
G	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
H	-	N.O.	-	-	-	-	-	-	-	-	-	-	-	-	-	-
IF	-	-	-	-	-	-	-	X	X	X	X	X	X	-	-	-
I	-	-	-	-	-	-	-	-	N.O.	X	X	X	N.O.	X	X	-

N.O.= Never Occur

CONTROL SYSTEM ARCHITECTURAL AND FUNCTIONAL DESIGN

As previously said, the control system has been designed by following a modular approach (Park et al. 2001). In particular, each Tern has an own control module, which communicates with the regarding Table, Island and Manipulator control modules (see figure 4). Moreover, each Tern control module is connected to the adjacent Terns control system modules, to coordinate the exchanges of semifinished shoes and lasts, and to the plant Supervisor that monitors the whole transport line.

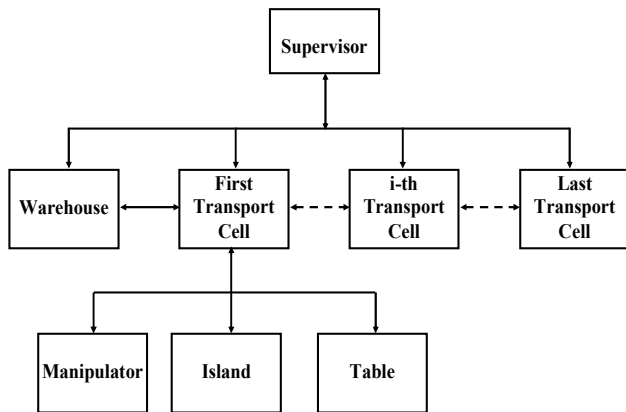


Figure 4: Molecular Line Control System Architecture

Each Tern control module is composed by two functional blocks: a “Supervisor” and a “Controller” (see figure 5). The Supervisor block is responsible for the scheduling action needed to coordinate the Tern itself with the adjacent Terns control systems, while the Controller block executes the opportune algorithms to carry out the operations on the semifinished shoes and lasts scheduled by the Supervisor block. Therefore, it communicates with the Table, Island and Manipulator control modules by means of proper signals.

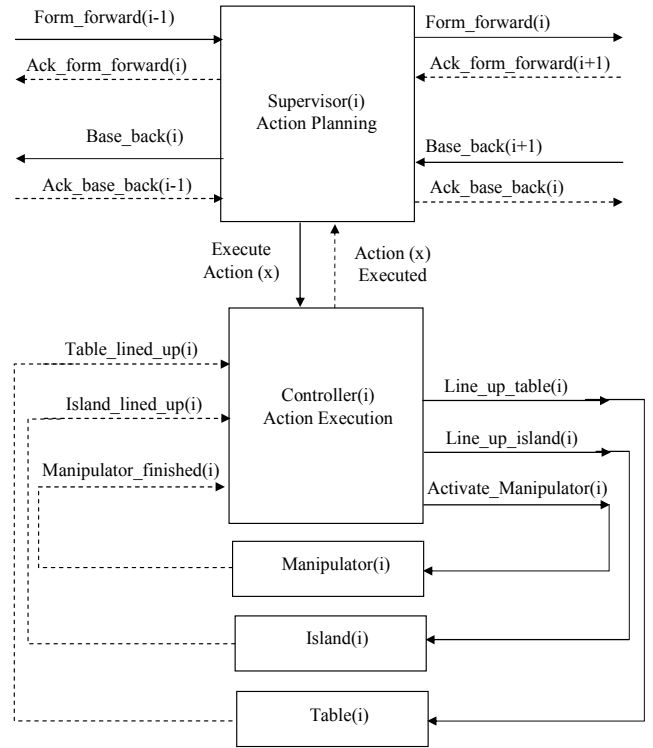


Figure 5: Tern Control Module Functional Blocks

Notive that the proposed modular approach allows to specify, design and verify only one Tern control system module, which can be used to control all the single Terns constituting the whole transport line. Such an approach is feasible thanks to the strong physical modularity of the considered molecular line, where the mechanical structure of all Terns is exactly the same one.

CONTROL SYSTEM IMPLEMENTATION AND SIMULATION BASED VERIFICATION IN ISAGRAF

The designed control system has been implemented in ISaGRAF by means of the Sequential Functional Chart (SFC) language, included in the IEC 61131 part 3 international standard (Standard IEC 61131-3° 1993; Lewis 1998). ISaGRAF is a CACSD (Computer Aided Control System Design) tool that allows structuring the software code (James et al. 1995), by means of all five IEC 61131 part 3 standard languages (ISaGRAF 1999). Moreover, it supports simulation functionalities for testing purposes, and automatic code generation facilities for different industrial PLC targets. The SFC language is a graphical formalism based on transitions and phases, to which Boolean conditions and actions are associated, respectively. The SFC formalism is suitable to describe discrete event systems, and has been directly derived from the Petri net formalism (David 1995). In figure 6 and 7 the hierarchical structure of the control software, and a detail of one of the SFC programs developed in ISaGRAF are shown, respectively.

obtained results show that the system is deadlock free and that the plant is well balanced, i.e. its resources are all used effectively.

CONCLUDING REMARKS AND FUTURE WORK

In the work the different phases of the design and simulation based testing of the logic control functions for an innovative agile shoe manufacturing plant have been discussed. In particular, the designed control system functions have been illustrated, and the closed loop discrete simulations performed to verify their correctness have been discussed, highlighting some related critical problems and the proposed solutions.

Future research activities concerning the presented work will address: a formal approach to the design of control system functions based on innovative approaches (e.g. multiagent or holonic techniques) and on emerging industrial standards (e.g. IEC 61499), and a more rigorous discrete simulation based testing method, in particular a more accurate plant model can be developed and suitable verification procedures can be defined in order to consider the different operating conditions and boundary conditions of the production plant.

REFERENCES

- Carpanzano E. and Ballarino A. 2002. "A structured approach to the design and simulation-based testing of factory automation systems", *Proceedings of the IEEE International Symposium on Industrial Electronics, ISIE2002*, L' Aquila, Italy, July 8-11, pp. 181-186.
- Renè David, "Grafcet: A Powerful Tool for Specification of Logic Controllers", *IEEE Transactions on Control system technology*, Vol. 3, N°3, September 1995.
- Ferrarini, L. and Carpanzano, E., 2002, "A Structured Methodology for the Design and Implementation of Control and Supervision Systems for robotic applications", *IEEE Transactions on Control Systems Technology*, vol. 10, no. 2, pp. 272-279.
- ISaGRAF, 1999, *Altersys*, Version 3.31F.
- James J., Cellier F., Pang G., Gray J. and Mattsson S.E. (1995). The State of Computer-Aided Control System Design (CACSD), *IEEE Magazine on Control Systems*, April 1995, pp. 6-7.
- R.W. Lewis "Programming Industrial Control Systems using IEC 1131-3" Revised edition 1998 *The Institution of Electrical Engineers USA*.
- Maffezzoni, C., Ferrarini, L. and Carpanzano, E. "Object-Oriented Models for Advanced Automation Engineering", *Control Engineering Practice*, n° 7, 1999, pp. 957-968.
- Park, E., Khargonekar, P., and Tilbury D. A Modeling and Analysis Methodology for Modular Logic Controllers of Machining Systems Using Petri Nets Formalism. *IEEE Transactions on Control Systems Technology*, Vol. 31, N° 2, May 2001, pp. 168-186.
- Standard IEC 61131, 1993. "Programmable controllers. Part 3: Programming languages", *IEC SC65B/WG7*.

BIOGRAPHY

EMANUELE CARPANZANO was born in Lörrach, Germany and received the 'Laurea' degree cum laude in Electronic Engineering and the Doctorate degree in Computer Science and Automation in 1995 and 1999, respectively, both from Politecnico di Milano, Italy. He has been a Visiting Scientist at the Technological University of Lund (Sweden) and at the European Space and Technology Centre in Noordwijk (The Netherlands). From 1999 to summer 2001 he has been Research Associate at the Department of Electronics and Computer Science of the Politecnico di Milano. He is currently Researcher at the Institute of Industrial Technologies and Automation of the National Research Council, where he is Responsible of the 2MaCS - Machine and Manufacturing Control Systems - research group. At present he is also Contract Professor of Automation at the University of Bergamo. In the last years he has been involved in different national and international research projects developed by universities, industrial companies and research institutions. He has published more than 50 papers on international journals and conference proceedings. His research interests include Advanced Motion Control Algorithms, Object Oriented Modelling and Simulation, Distributed Industrial Control Systems Development Methodologies and Tools, Hybrid Systems, Symbolic Manipulation of Differential Algebraic Equations Systems and Gain Scheduling Control Strategies.

ANDREA CATALDO was born in Milan, Italy and received the 'Laurea' degree in Electronic Engineering in 1995 from Politecnico di Milano, Italy. From October 1996 to May 1998 he has been Instrument Engineer in Foster Wheeler Italiana, a leader Engineering Company, where he has worked on Distributed Control System base engineering. From June 1998 to November 2002 he has been Researcher in ABB Corporate Research Italy and he has worked on Industrial Power Plant Simulator for the Testing of Distributed Control Systems and on Medium Voltage Electric Power System Protection Devices. He is currently Researcher at the Institute of Industrial Technologies and Automation of the National Research Council, where he is working on analysis, design, implementation and testing of logic control systems. His research interests include Object Oriented Modelling and Simulation, Distributed Industrial Control Systems Development Methodologies and Tools and Gain Scheduling Control Strategies.

PRODUCTION-BASED PROFITABILITY SIMULATIONS IN PAPER MILLS

John Fogelholm
D. Sc.(Tech.), Ph.D.,
Helsinki University of Technology,
Department of Industrial Engineering and Management,
Finland.
E-mail: John.Fogelholm@hut.fi

KEY WORDS:

Economic steering applications, product costing, economic simulations in paper mills

ABSTRACT

The basic usefulness of any mill wide, economic steering software system, including its simulation aspect, and specially developed for paper mills, resides in the range, accuracy and cross-referencing of information, required by mill management on different hierarchical levels in their everyday activities. This means in practice that the different resource consumptions, included in forecasted individual production runs, can be changed, and the impact simulated as required in different ways, and the calculated results expressed in monetary terms. This pertains particularly to the product costing and profitability aspect of the total potential product range of any individual mill.

1. INTRODUCTION

The basic information for decisions concerning future activities in paper mills are based either on strictly historical data from previous production activities, or based on the anticipated behaviour, expressed in either resource consumption or economic terms. The requirements for an accurate enough steering system, and the corresponding parts of an adequate computer program, differ thus substantially from each other in these two cases. See Figure 1.

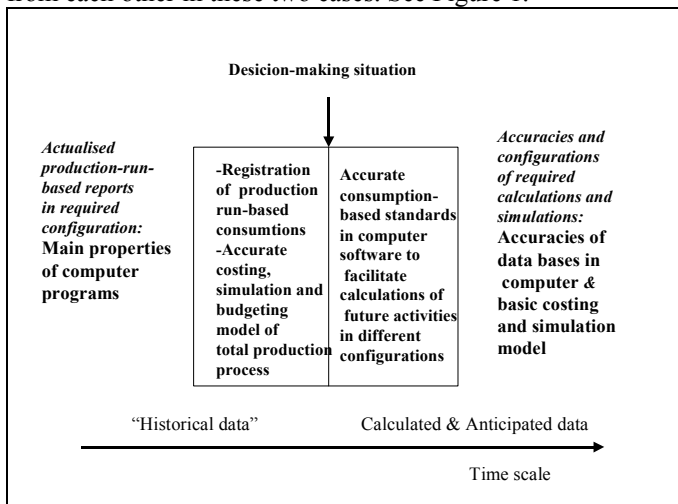


Figure 1: Computer Program Basics

There are two main considerations concerning the treatment and utilization of basic historical data. The first concerns the availability of production run-based resource consumption information from different databases, and the other concerns the required properties of the computer program to generate reports in almost any required combination.

As the amount of basic information required and utilized in the Ex Ante economic steering calculations is tremendous, the required simulations have to be performed in real life with an accurate enough computer program, which will be dealt with in this paper.

2. BASIC HIERARCHY IN MILL REPORTING

Any accurate enough mill-wide economic steering system, used both for Ex Ante and Ex Post purposes in paper mills has to be based on the resource behaviour of the individual production runs, as shown in Figure 2. From these figures can them be calculated and combined different reports, both concerning the required combination of individual production runs, and the corresponding results on the level of individual paper machines or production paths, and the individual mill itself.

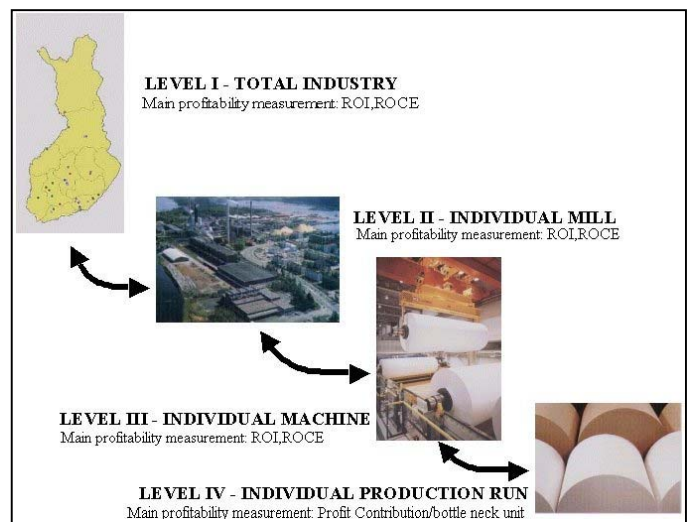


Figure 2: Hierarchical Levels in Paper Mill Activities

3. SIMULATION PROPERTIES OF A STEERING TOOL

The fundamental requirement for forecasting purposes, depicted on the right side in Figure 1, and basis for any simulation activities, consists of their ability to anticipate, forecast and determine the resource requirements for individual orders, which mostly differ only incrementally from each other. This is the situation in the typically divergent production process for paper products, as far as raw material content, dimensions of the final product, and quantities to be produced (Fogelholm2000a).

The most important aspect of any industrial application of systems dealing with forecasting/predictions, has to be considered its ability to predict as accurately as possible the resource or economic behaviour of the process under investigation (Fogelholm1999a). The scientific tool that is probably the most useful in judging the accuracy or calculating the complexities of the various input elements of any industrial process the systematic treatment of these inputs

in the form of a model, and thus also of the final output, is the application of variety analysis to any production process under scrutiny, in this the production process of a paper mill. This usually includes the application of regression analysis to the basic data required. Variety, in this context, is defined as the number of possible states of the variable whose complexity we want to measure (Beer 1990). This means that this anticipatory aspect of the actual use of a system, consisting of different input elements and the basic model that converts this data into output information, represents the main yardsticks of its concrete usefulness. The problem of the inherent variety in each individual resource consumption, and its impact on its actual value to be used, is clearly seen in Figure 3, which shows the actual throughput values for different products on a paper machine, and the final values to be used in calculations.

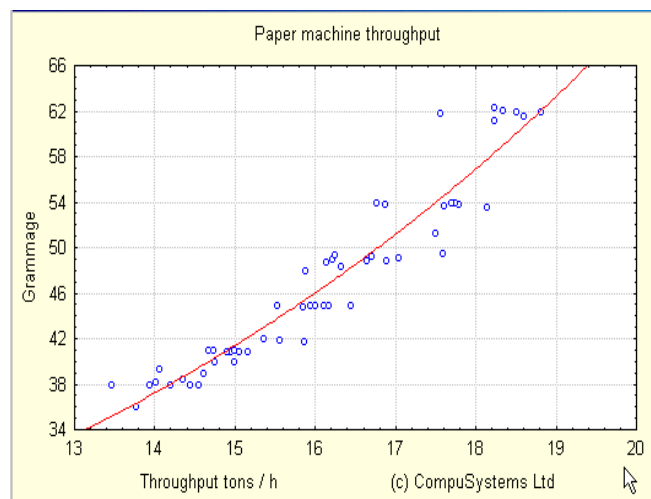


Figure 3: Throughput values for total production range

4. OTHER PREREQUISITES FOR ECONOMIC STEERING ACTIVITIES

One most important prerequisites for simulation activities, concerning the production of different grades in a mill, consists of the requirement of the model to anticipate the individual resource behaviour for the total range of products, which can be manufactured, including the determination of the production path chosen in each case (Fogelholm 1997). These basic requirements are shown in Figure 4 (Fogelholm 1997).

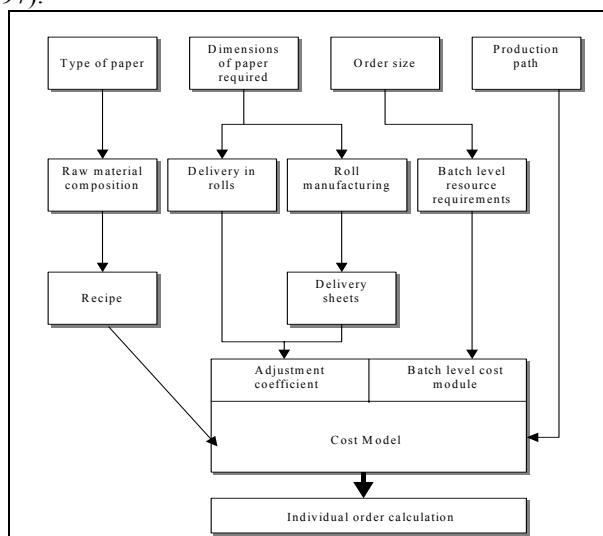


Figure 4: Basic information requirements for simulation

All the above-mentioned basic data are required to determine accurately the required resource consumption in the individual production run. This represents the 'foundation', on which the different calculations required are to be based. (The adjustment coefficient mentioned in the cost model represents the impact on the individual resources, connected to recycled trim- and sheet waste).

The optimal steering levels in paper mills, from an economic measuring point of view, are thus found to be as follows (Fogelholm 2002a):

- Individual production run
- Production line (production path)
- Individual mill

5. BASIC CALCULATION OF INDIVIDUAL PRODUCTION RUN

The first main application of any computer program, which concerns product costing and its applications, consists of calculating the actualised result of, or simulating the economic result of individual production runs of potential orders on different production paths, taking into consideration the impact of alternative order sizes and product characteristics, as for instance the weight of the product to be manufactured, and the required final dimensions. The result for both situations is shown in the form of these required product calculations in Figure 5. The calculation form and the groupings of individual cost elements naturally have to be customized, according to the requirements of the individual mill.

Scan pricing calculations \$/tn			
Paper Mill	KON5	Raw materials \$/tn	1,186.41
Gsm	52	Auxiliary materials \$/tn	43.43
Order quantity	1,000	Fillers \$/tn	54.51
Unit price	4.00	Paste \$/tn	0.00
Date	28/10/1997	Profit contribution 1:	2,715.65
Calc nr	Type	Order nr	Customer
1	105200	1000/0000/00/0	NIPPON P
2	116000	2000/0000/00/0	VOGUE
4	105800	6000/0000/00/0	NUEVA YC
		Profit contribution 2:	2,135.89
		Supercalander	20.83
		Coater	0.00
		Prewinder 1	0.00
		Prewinder 2	0.00
		Slitter winder	35.00
		Packaging	45.00
		Warehouse	0.00
		Dispatching	44.00
		Profit contribution 3:	1,991.06
		Paper Machine fixed costs	3.91
		Common PM fixed costs	23.48
		Factory fixed costs	215.26
		Depreciation	78.28
		Interest	58.71
		Profit / Loss	1,611.42
		Profit contribution / machine hour	15,592.00

Figure 5: Example of production run economic calculation

The basic model has also to take into consideration the calculations of different trim, sheet and other waste, and to calculate their economic impact, together with all main cost elements of the required set-ups, or grade changes, as indicated in Figure 5.

6. MAIN APPLICATIONS OF SIMULATIONS

The main practical applications of the predictive aspect of these kinds of simulation models and calculation systems are: calculating the economic result of potential and actual production runs on different alternative production paths, the impact of alternative individual resource consumption inputs creation and maintenance of making resource budgets for future time periods, during which the breakdown of the product range are to be determined.

The latter application is of apparent economic significance, as it saves a lot of time for managers in charge of budgeting. This particular calculation is shown in the screen shot of Figure 6, where the total consumption is calculated, based on the individual calculations of the product runs included. The same screen shot is used for showing calculated material budgets both for the total mill and for the individual machines, for the different production runs, for different customers, for different countries and for different types of products, etc. The same information can thus be used for showing both future and realized total material consumption. Figure 7 shows the impact of changes in raw material prices on the profitability of an individual production run.

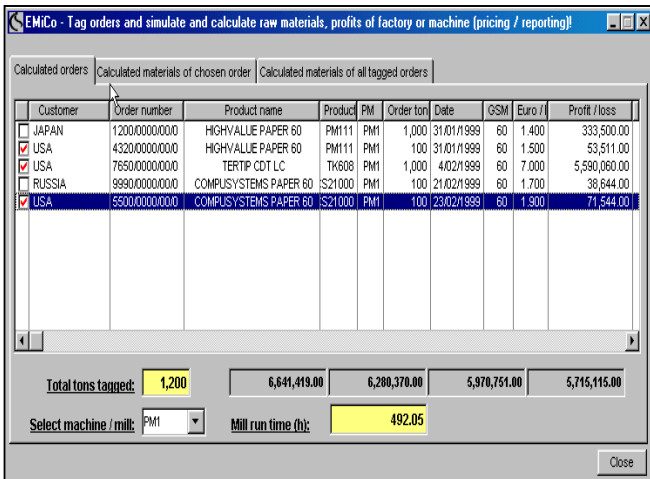


Figure 6: Resource consumption-based budgetary simulation and calculation

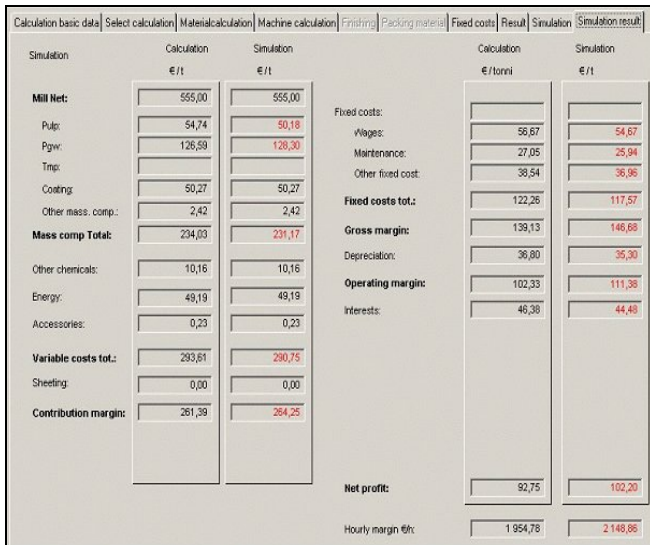


Figure 7: Simulation of changes in raw material prices and other resource components

The calculations of predictive results discussed above represent different aspects of simulation activities, which are required by paper mill management.

An accurate industrial simulation model and the corresponding computer program can also be utilized to make comparisons of the impact of different production elements on the profitability of the total mill, the individual paper machine, the individual product, or the whole production process. A typical example of this is the situation where a material supplier, like those of wires and retention chemicals, claims that his products have superior characteristics, which could be utilised to increase the profitability of the mill. In a case like this, the economic impact of this new product can easily be assessed.

7. "MILL-WIDE" OR "STAND ALONE" INSTALLATIONS

A mill-wide total integrated economic steering system discussed above, requires a substantially greater investment in the required interfaces with the required data files in Figure 8, compared with a corresponding "Stand-alone"-version for simulation, pricing and budgeting purposes.

Any mill, slightly hesitant about whether to implement directly a mill-wide system, should consider starting with a "Stand-alone"-development phase.

The system outlined above, DNAemico, incorporating the scientific requirements outlined above, is in actual use in several Finnish paper mills and one company's headquarters since 1997. All of the actual mill-installations are of the Mill-wide type.

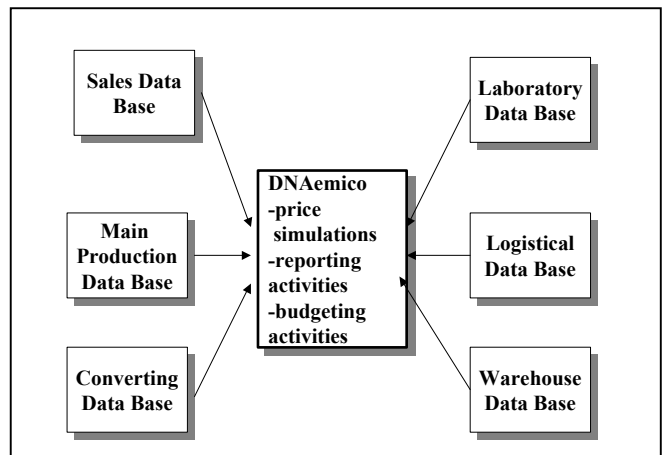


Figure 8: Required reporting and simulation configurations.

8. PREVIOUS SCIENTIFIC REPORTS OF THIS INDUSTRIAL SIMULATION TOOL

The simulation tools, developed and applied for different industrial simulation purposes have to be both industry and application specific. The basic resource consumption model developed for profitability simulation purposes, which represent one important aspect of a total economic steering computer program, has already been reported both in a previous simulation conference (Fogelholm2000b) and in a Paper Industry specific conference (Fogelholm and Hamalainen 2002). The usefulness of this kind of an actual profitability optimisation tool has been tested and verified in Finnish paper mills. The actual computer programs have been developed, based on the experiences of the writer of this

paper, together with personnel in Finnish paper mills, working together with computer programmers.

9. ARCHITECTURE

DNAemico (Economic Mill Control) is a MS-Windows NT based software product and it is now a part of the Metso Automation's MetsoDNA (Dynamic Network of Applications) product family. Basic information for the required calculations in a mill-wide system can thus be gathered by on-line interfaces from a variety of systems and data bases, including sales and order handling, accounting and invoicing, material management, production and recipe management systems.

The shared pool of information is managed in a flexible relational database in a client-server network. With DNAemico it is easy to configure even complex production paths consisting of rewinding, coating, calendering, sheeting, packaging and transport.

The final screen shots and configurations in actual mill applications will naturally have to be tailor-made, owing to the specific requirements of the individual mill. The basic configurations are however, basically similar to the one shown above.

REFERENCES

1. Fogelholm, J.2000. *Cost Function Modelling in the Paper Industries*. Report No.11. Helsinki University of Technology, Department of Industrial Engineering and Management. 2000. ISBN 951-22-5009-8,
2. Fogelholm, J. 1999. *The Utilization of Cybernetics in the Steering of Complex Production Processes*. Computing Anticipatory Systems: CASYS'98 - Second International Conference. Ed.: D. Dubois. CP 465, American Institute of Physics. ISBN 1-56396-863-0, 543-554
3. Beer, S. 1990. *The Hearth of Enterprise*. John Wiley & Sons. ISBN 0-471-27599-9
4. Fogelholm, J 1997. *Product Costing in the Paper Industry*. Journal of Cost Management. 11/5, 42-46.
5. Fogelholm, J. 1999b. *Cost Models for the Paper Industry*. Paper Asia. 15(1). 27-30.
6. Fogelholm, J 2002. *Economic Steering on Different Hierarchical Levels in Paper Mills*. Paper Technology. PITA 43/4, 51-54.
7. Fogelholm, J 2000. *Anticipated Resource Consumption Modelling in the Production Process as the basis for Simulation Activities in Paper Mills .Proceedings of 14. European Simulation Multiconference. Ghent. SCS, 331-334.*
8. Fogelholm, J & Hämäläinen, E. 2002 *Economic Reporting in Different Configurations in Paper Mills-DNAemico*. ATIP: 55. Congress Annuel. Bordeaux. p.5

BIBLIOGRAPHY

John Fogelholm, D.Sc.(Eng.) 2000, Ph.D. 1993, Lic.Tech. 1979, M.Sc. (Eng) 1963, M.A.(Pol. Sc.) 1964.

He has been, during his previous work in connection with the paper industry, manager in charge of economic development in Finland of Nokia paper mill, and of Rosenlew paper and packaging production. He has also been working as a UNIDO expert in paper mills and other type of factories abroad. During a 10-year period he was also working as Industrial Councillor, Scientific Attache' and Commercial Secretary in Bruxelles, Tokio and Abidjan.

Presently he is teaching at the Helsinki University of Technology since 1988, where his speciality is Cost and Profitability Management in the paper industries. He is also Visiting Professor at AIT (Asian Institute of Technology) in Thailand, where he regularly once a year teaching the subject of Economic Steering of paper mills. He is the author of some seventy articles on cost management and technology transfer, has written and presented 35 International Congress papers on Costing and Economic Steering Systems in general, and their applications to paper mills in particular, and has written 3 course books on cost management, both in Finnish and in English. His articles have been published in 10 different languages.

He has been heavily involved with the development of mill-wide, computer-based models for the economic steering of paper mills, which are already implemented in Finnish paper mills.

PERFORMANCE ENHANCEMENT USING AN EXPERT MECHANISM IN A MANUFACTURING SIMULATOR

Habtom Mebrahtu

Rob Walker

School of Design & Communication Systems

Anglia Polytechnic University

Victoria Road South, Chelmsford

Essex CM1 1LL, United Kingdom

H.Mebrahtu@anglia.ac.uk

r.w.walker@apu.ac.uk

Tony Mileham, Professor

Department of Mechanical Engineering

University of Bath

Bath, BA2 7AY

United Kingdom

A.R.Mileham@bath.ac.uk

KEYWORDS

Manufacturing Simulation, Optimisation

ABSTRACT

The need to include a mechanism that could assist in analysis and performance enhancement of simulation models has been under discussion for a long time. Many simulation packages on the market offer powerful “what-if” evaluation techniques for production planning. However, most of them rely on the user's own experience to interpret the results after each simulation, and anyone without such experience would find it difficult to make reasonable sense of the results before deciding on the next simulation run. This paper describes the use of an expert mechanism that could be integrated into a simulation package to facilitate the process of interpreting and assessing simulation results and in improving performance. It also discusses the need for checking stability of the model before reporting the model as realistic.

INTRODUCTION

Although simulation in manufacturing has traditionally been used for high level capacity planning, there are many other benefits in using simulation. Factory layout, production routing, production mix, and throughput prediction, bottleneck identification, new resources deployment, to name but a few, can all be predicted using simulation. Ben-Arieh, while explaining the need of simulation asserts that in modern manufacturing facility, the available flexibility introduces another degree of flexibility in decision making. The lack of clear understanding of the dynamics and interaction of components of modern manufacturing systems calls for the use of simulation as an essential support tool. Simulation is no more a niche management tool, which can only be afforded by a few, thanks to ever increasing computer power and its affordable price. The advancement in programming and software engineering also means that very clever simulation software has hit the market, with highly configurable user features and powerful animation [Mebrahtu & Lung].

However, these powerful features are generally focused on the front-end of creating a manufacturing model easily and on

getting simulation results quickly [Pengen et al]. As a result, massive reports, which include statistics, tables and a lot of raw data are generated, but do not help the user see the connection of these reports to the next appropriate action in a consistent and logical way. Any interpretation and action will depend solely on the user's experience in using simulation. Additionally, limited alternative simulation models could be dealt with in a traditional way but as the possible alternatives increase, conducting a large number of simulation runs becomes time consuming and costly (Morito et al).

Some commercial simulation packages now include some type of integrated optimisation routine, Optimiser in Witness and OptQuest in Delma [Fu et al], for instance. The goal of these routines is to seek improved settings of user-selected system parameters with respect to the performance measure(s) of interest. However, unlike mathematical programming packages, there is no way of knowing that an overall optimum has actually been reached, thus *optimisation* may be a loose word.

The experimental work in this paper illustrates the use of a rule-based algorithm that is integrated with a simulation package to analyse simulation output, assess performance of a production floor, and automatically change the controllable variables within given constraints to enhance performance. Once the stability of the original model is checked, each time the simulation is executed, the rule-based algorithm would interpret and analyse the results, and suggest a suitable action plan for the next iteration for further improving the performance. Such a concept also opens up a huge possibility of running the rule-based simulator remotely across the Internet, hence allowing smaller companies to benefit from simulation.

EXPERIMENTAL MODEL

Base Model

In order to demonstrate how simulation results can be translated into action plans, and how different production scenarios can be compared using performance indices, a case study factory model with limited operation and resource flexibility has been set up as shown in Figure 1. The experiment was based on a company model obtained from the Lanner Group, the software house behind the

WITNESS simulation modelling system. Some operational data have been modified for simplicity.

At the start of the experiment, the case study company experienced a severe backlog in sales orders due to antiquated machinery and poor production planning. Assuming that there was a demand for up to five times of the current product output, a series of simulation runs were set up to evaluate the effect of investing appropriate resources against the possible increase in throughput and benefits.

The model consists of seven main operations. The manufacturing process starts with the stock of bars that come into the saw area stock buffer. The bars are then cut producing 3 blocks from each bar. After sawing, the blocks go to a belt conveyor that transfers the cut bars to the coating operation. The coating machine coats 6 blocks at a time. Once coated the blocks are placed in the staging area adjacent to the inspection station. The inspectors then determine the quality of each block's coating and send it either to hardening, or to the rework buffer. The hardened blocks are then loaded into special fixtures so that four blocks can enter a grinder at once. There are two grinders available with no priorities between them. Once ground, the fixture and the four blocks are placed into an unloading station where the blocks (now valves) are sent to the finished stock areas and the fixtures onto an overhead conveyor. The conveyor puts the fixtures back into the fixture buffer for reuse by the loading machine. Witness was used to model the system.

Model Stability

It is important to ensure that the model is not significantly affected by changing the random number streams. If it does, then the model results cannot be expected to give a solution that would be realistic. The stability of the model was checked by conducting 25 runs with different random number streams for each run and for each element and each corresponding data. This was meant to give the feel that events were following more realistic randomness. Outputs of the 25 runs were recorded and a Cusum (90% confidence) conducted as shown in Figure 2. The mask used is a C2 semi-parabolic mask as defined in BS 5703 part 3. It could be seen from the graph that the data were all between the upper mask and the lower mask, indicating good consistency. A similar stability check was conducted on the last model and it again showed satisfactory stability.

The reason for selecting Cusum charts to check stability is to allow future development of the program to automatically check that the change being proposed will result in a significant improvement. The development process will continue by integrating an 'evolutionary operation' (EVOP) design of experiment system into the program, which will be controlled and monitored by Cusum charts [Walker], until an optimum solution is achieved.

THE RULE-BASED EXPERT MECHANISM

Objectives

The existing system can manufacture around 144 valves every 75 hours. It has been established that the benefits of an increase in throughput by 1 valve can be fully justified for an investment of £250. That is, for each investment of £250, there must be an increase of at least one part. A maximum amount of £75,000 is available to be spent for the investment, which amounts to an equivalent of 300 more valves to justify the spending. The main investment costs expressed in terms of production benefits are shown in Table 1. Each item has been assigned a cost equivalent in parts.

Methodology and Results

As previously described, our main performance index is net profit (or net saving) which is the difference between the increase in throughput and the investment (expressed in terms of equivalent parts). The main rules used include the techniques of Theory of Constraints and line balancing backed by concurrent monitoring of investment. It involves mainly identifying bottlenecks and blockages that are used as the basis for actions to be taken in each sequential simulation run.

Witness as an object link embedding (OLE) automation server could be controlled by Visual Basic (VB) which is an OLE controller [Lanner Group]. Relevant input/output data to Witness as well as running of Witness could be controlled with VB (with some assistance from Excel). Therefore, using VB to develop the expert mechanism was ideal. The simulator uses data displayed in Excel but controlled by VB, runs the model and generates output. The expert mechanism receives the relevant output data from the simulator, manipulates the data, assesses model performance and generates recommended changes for the next run. The iteration goes on until a limiting factor is reached, at which point the result would be output.

Eleven simulation runs were conducted with the summary of results shown in Figure 3. The results indicate that all except runs 4 and 7 could be justified for their respective investments. Models 10 and 8 showed the better net savings, with model 10 significantly favoured both in savings and its throughput in view of rectifying the current problems of the company.

CONCLUSION

Although the model is limited in many respects, it highlighted the basic concept of integrating an optimising element to a manufacturing simulator for automatic results analysis and performance enhancement. Various performance assessment methods such as throughput, inventory level, machine utilisation and investment can be incorporated into future experiments to make the system more versatile for a wider spectrum of simulation scenario. The proposed concept can handle a mix of different production objectives whereby users can set target figures

with each objective, and the system will iterate until those targets are met within specified allowance.

In an ever growing popularity of the Internet, making the system WEB compliant is another goal in future research. When fully developed, it is possible that registered users from remote sites can use the system by providing required inputs to the simulator, target objectives, constraints of scenarios, plus other necessary details required to build and run a totally customised model on the net. The simulation

system will then run continuously at the host web site until the targets and constraints are satisfied. The remote user can then view the optimised results and the accompanying conditions. This concept of **Application-On-Demand** (AOD) has yet to be materialised but has great potential in allowing smaller firms to benefit from specialised application software such as manufacturing simulation, with the consent of the software suppliers.

REFERENCES

- Ben-Arieh, 1986, *A Knowledge-based system for Simulation and Control of FMS*, p 287-295, IFS Publications Ltd.
- BS 5703 Part 3 (1981)
- Fu M C et al, 2000, "Integrating Optimisation and Simulation: Research and Practice", *Proceedings of the 2000 Winter Simulation Conference*, p610-617,
- Mebrahtu M, Lung A, 2001, "Prediction of Shop floor Performance using a flexible manufacturing simulator ", *World Manufacturing Congress 2001*, ICSC-NAISO Academic Press.
- Morito S, Koida J et al, 1999, "Simulation-based constraint generation with applications to optimisation of logistic system design", *Proceedings of the 1999 Winter Simulation Conference*, Edited by Farrington, Nembhard, Sturrock & Evans.
- Lanner Group, *OLE Automation in Witness, Witness 2002 on-line*, Lanner Group, 2002.
- Pengen C D, Shannon R E, Sadowski R P, 1990, *Introduction to Simulation*, Mc Graw-Hill.
- Schormann F, Perera T, 1998, "Model representation methods in simulation of manufacturing systems", p292-296, *International Conference on Simulation 98*, York University, UK.
- Walker R W, 2002, "Evolutionary Operation (EVOP) Evaluated by Cusum Charts", *Conference paper at World Manufacturing Congress*, ISBN 3-906454-28-2.
- Witness training course notes from Lanner Group, 1998

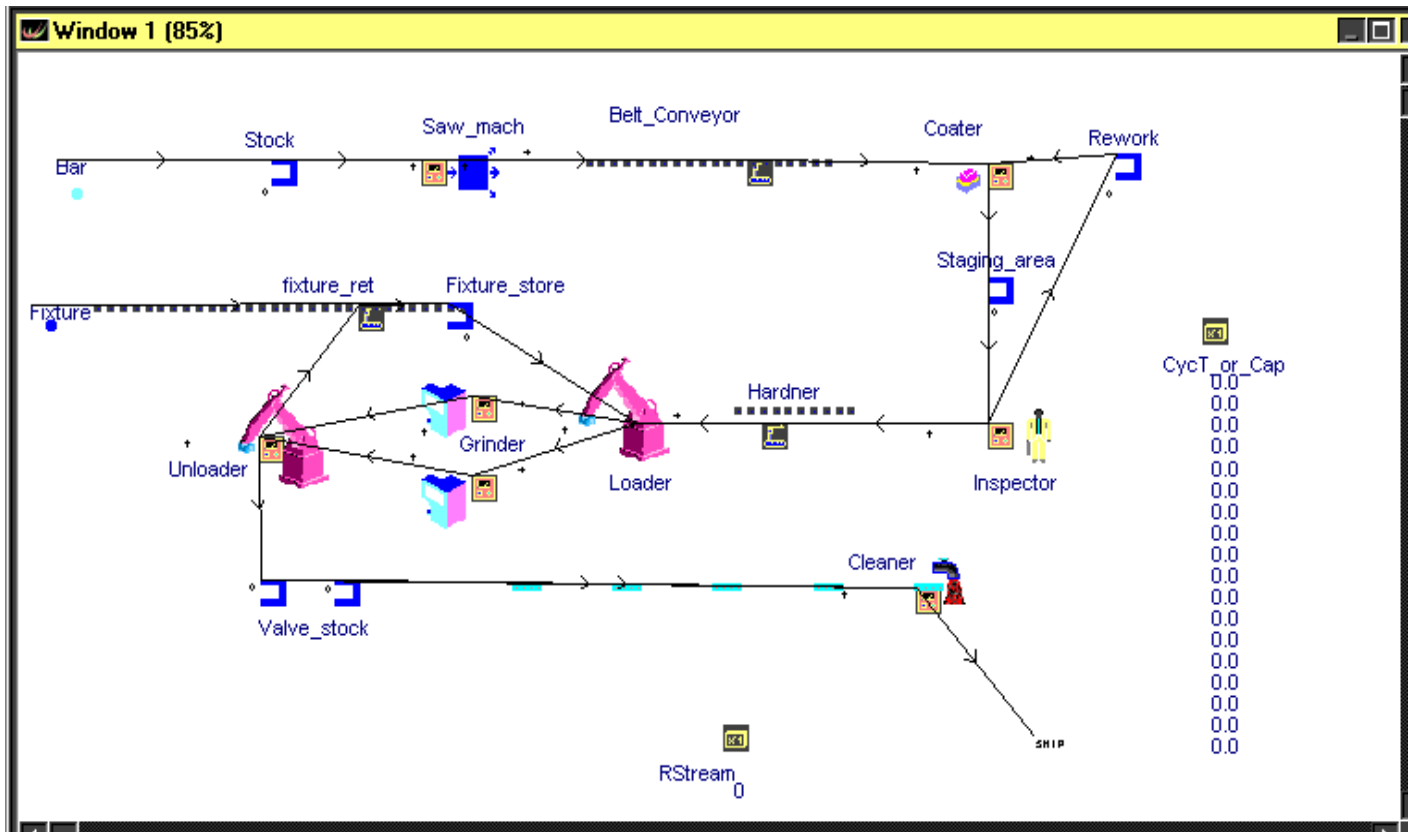


Figure 1 – The simulation model used in the experiment

Element	INVESTMENT COST-EQUIVALENT IN PARTS		
	New Element	Decrease cycle time by 10%	Decrease set-up time by 10%
Buffers	0.2		
Saw_machine	100	10	20
Conveyor		10	
Coating machine		24	
Inspector	80	24	
Hardener (Furnace)		24	
Loader/ Unloader	64		
Grinder	200	40	
Cleaner	40		

Table 1 - Possible Costs for Investment

Run No	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Valves, x	136	145	144	144	148	148	140	148	144	145	150	152	156	154	140	148	148	152	154	148	148	144	152	147	140
x-T	-11	-2	-3	-3	1	1	-7	1	-3	-2	3	5	9	7	-7	1	1	5	7	1	1	-3	5	0	-7
CuSum	-11	-13	-16	-19	-18	-17	-24	-23	-26	-28	-25	-20	-11	-4	-11	-10	-9	-4	3	4	5	2	7	7	0
Upper mask															41.86	39.4	36.93	34.47	32.01	29.55	27.08	23.14	17.73	11.33	4.924
Lower mask															-41.9	-39.4	-36.9	-34.5	-32	-29.5	-27.1	-23.1	-17.7	-11.3	-4.9

Target, T	147
StdDev	4.924

Ctrl Factors	8.5	8.0	7.5	7.0	6.5	6.0	5.5	4.7	3.6	2.3	1.0
--------------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

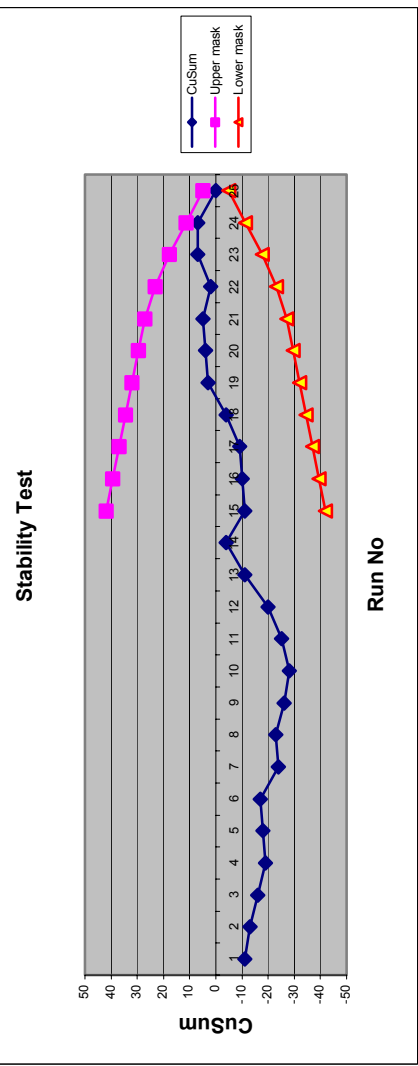


Figure 2. Stability Test (CuSum Test)
Initial model

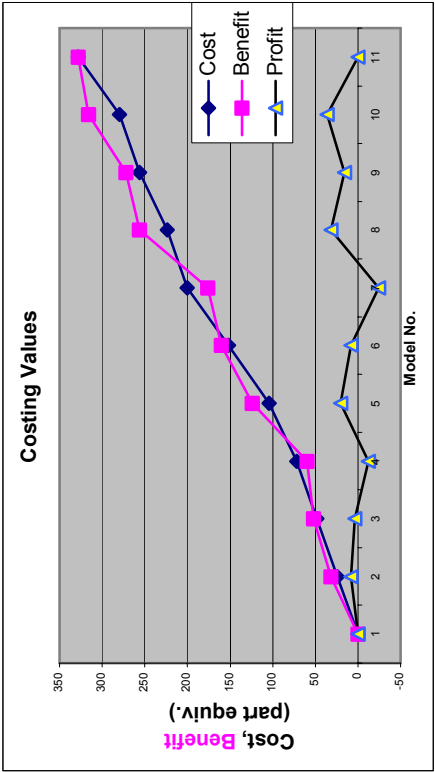


Figure 3. Simulation Results and Costing

Model No.	1	2	3	4	5	6	7	8	9	10	11
Valves shipped	136	168	188	196	260	296	312	392	408	452	464
Cost	0	24	48	72	104	152	200	224	256	280	328
Benefit (increase in parts)	0	32	52	60	124	160	176	256	272	316	328
Profit (Saving)	0	8	4	-12	20	8	-24	32	16	36	0

Manufacturing Integration

AN INTERFACE DATA MODEL SUPPORTING MANUFACTURING SIMULATION

Y. Tina Lee

Yan Luo

Chuck McLean

National Institute of Standards and Technology

100 Bureau Drive, MS 8260

Gaithersburg, MD 20899-8260, USA

E-mail: leet@cme.nist.gov, luoyan@cme.nist.gov and mclean@cme.nist.gov

KEYWORDS

data exchange, information model, machine shop operations, simulation

ABSTRACT

Simulation technology is underutilized by the manufacturing industry. Standard data interfaces, among simulation modules or between simulation and other software applications, could make information sharing effective and efficient, and hence promote utilization of simulators. An information model that represents machine shop data and facilitates data sharing among the manufacturing execution system, the scheduling system and the simulation system modules in a machine shop has been under development at the National Institute of Standards and Technology (NIST). A set of relational database tables representing all elements in the information model is being developed. The model will be used to support the integration of software applications at a pilot machine shop located in Pittsburgh, PA. The information modeling project is part of efforts that support the development of standard data interfaces. This paper presents an overview of the information model, the mapping from the information model to relational database tables, and the future project plan.

1 INTRODUCTION

Today's industry is characterized by increasing pressure on development time, manufacturing cost, and product quality. In manufacturing industry, simulation technology can reduce cost, save time, and improve quality. A manufacturing system is composed of a large number of distinct processes that all influence product cost, product quality, and productivity. The interactions between these various facets of a manufacturing system are complex, and decisions made concerning one aspect have ramifications

that extend to the others (Stoll 1991). A major reason that an enterprise cannot respond to a sudden change is lack of fundamental thinking about the relationship between management and information in the manufacturing enterprise (Inoue et al. 2000). Information technology is critical to enterprise integration. Formal information modeling languages that describe information requirements unambiguously are an enabling technology that facilitates the development of a large scale, networked, computing environment that behaves consistently and correctly (Lee 1999).

A machine shop data model, as an interface information model, has been under development at NIST to support both NIST's System Integration of Manufacturing Application (SIMA) program (Carlisle and Fowler 2001) and the Software Engineering Institute's Technology Insertion Demonstration and Evaluation (TIDE) program (McLean et al. 2002). SIMA supports NIST projects applying information technologies and standards-based approaches to manufacturing software integration problems. The goal of the TIDE program is to demonstrate the cost-saving and efficiency benefits that small manufacturers can gain by using advanced commercial off-the-shelf software and information technology. TIDE focuses on helping firms that supply goods and services important to national defense; however, much of the work of TIDE is broadly applicable to small businesses.

Information integration and sharing require the support of a united and neutral information model (Luo 2000). Some languages such as the Integrated Computer Aided Manufacturing (ICAM) Definition Language 1 Extended (IDEF1X) (D. Appleton Company 1985), EXPRESS (ISO 10303-11 1994), Unified Modeling Language (UML) (Object Management Group 2003) and Extensible Markup Language (XML) (The World Wide Web Consortium 2000) are most frequently used by the manufacturing enterprise for information modeling. IDEF1X is a formal graphical language for relational data modeling, developed by the U. S. Air Force, and is more popular in the USA. EXPRESS was designed to meet the needs of

the STandard for the Exchange of Product model data (STEP) (ISO 10303-1 1994), it has been used in a variety of other “large-scale” modeling applications. UML is a graphic representation for artifacts in software systems, and is also useful for database design. XML is a format for structured documents, and it helps make possible information exchange in a globally distributed computing environment. NIST’s machine shop data model is developed using UML and XML. The model is currently being implemented using XML documents and relational database tables. XML is chosen for supporting web users while relational databases are commonly used by today’s small manufacturers.

2 OVERVIEW OF THE MACHINE SHOP DATA MODEL

NIST’s machine shop data model is presented in both graphical form, in UML, and textual form, in XML. The model has been developed with two goals in mind: a) the model will be used to support the integration of software applications at a pilot facility, the Kurt L. Lesker Company’s machine shop, and b) the model will be promoted as a standard data interface for manufacturing simulators and possibly other software applications. The model contains twenty major elements. The name and brief description of the major elements are described as follows.

- The element *organizations* is used to maintain the organizational structure, contacts, and address information for the manufacturing organization, and its customers and suppliers.
- The element *calendars* identifies the shift schedules that are in effect for a period of time, breaks, and holidays.
- The element *resources* describes all the resources that may be assigned to tasks in the shop. The resource types available in the machine shop environment include: stations and machines, cranes, employees, and tool and fixture catalog items.
- The element *skill-definitions* lists the skills that an employee may possess and the levels of proficiency associated with those skills.
- The element *setup-definitions* typically specifies tool or fixture setups on a machine. Tool setups are typically the tools that are required in the tool magazine. Fixture setups are work-holding devices mounted on the machine. Setups may also apply to cranes or stations.
- The element *operation-definitions* defines the operations that may be performed at a particular station or group of stations in the shop.
- The element *maintenance-definitions* defines preventive or corrective maintenance to be done on machines or other maintained resources.
- The element *layout* defines the locations of reference points within the shop, area boundaries, paths, resource, and part objects.

- The element *parts* provides attributes for part specifications, group technology codes, customers, suppliers, as well as links to bill of materials, process plans, drawings, part models, and other references.
- The element *bills-of-materials* cross-references the parts and quantities required in a hierarchical bill-of-material structure.
- The element *inventory* identifies the instances and locations for the part, material, tool and fixture inventory.
- The element *procurements* identifies the external purchases that have been created to satisfy the part inventory and manufacturing requirements.
- The element *process-plans* specifies a set of process plans that are associated with production and support activities for a particular part or parts.
- The element *work* is used to specify a collection of a hierarchy of production orders, jobs, and tasks. It is also used to specify a collection of internal support orders for maintenance activities, inventory picking, and tool preparation.
- The element *schedules* lists the planned assignment or mapping of work to resources and resources to work.
- The element *revisions* specifies information about a set of revisions of the subjects. Information included in *revision* are each revision’s identification, a description, the date, the creator, etc.
- The element *time-sheets* provides a list of individual *time-sheet* elements. A *time-sheet* is used to log the hours that an employee works, the time the employee takes off from work, and the accrual of leave hours.
- The element *probability-distributions* specifies the distributions that are used to vary processing time, breakdown and repair time, availability of resources, etc.
- The element *references* is used to express information about reference materials that support or further define the data elements contained within the shop data structure.
- The element *units-of-measurement* describes various measurement units used in the file such as for distance, speed, mass, time duration, currency, etc.

3 THE DATABASE MODEL

A database provides the possibility of data sharing in a distributed simulation environment. A set of relational tables that represent the elements within the machine shop data model and the relationships among elements are being developed.

3.1 Database structure design

A database model is designed to map onto the XML documents. The database contains a set of relational tables presented in a tree shape structure. Table 1 is a design view of the data structure of *shop-data*. The design

view contains three types of information presented in three columns: Field, Data Type, and Description. The “Field” column identifies attributes of the element and special elements that are used by the database system, or the referenced or referencing tables. The “Data Type” column defines the data type format for the field. The “Description” column is reserved for information about the attribute usage, domain, definition (that defines the internal relationship between tables), reference, cardinality relationship, etc. The cardinality relationship specifies how many specific instances of the child element could be related to the parent element. The cardinality relationship may be one to zero or one, one to zero or more, one to one or more, or exactly “n” occurrences, and is presented in the design view as [0: 1], [0: *], [1: *], [n], respectively.

Table 1: Data Structure for *shop-data*

Field	Data Type	Description
INDEX	Number	[1],system use
type	Text	[1],type.type-category = "shop-data"
identifier	Number	[1]
number	Text	[1]
name	Text	[0..1]
description	Text	[0..1]
REFERENCE-KEYS	Text	[0..1],reference-keys.SOURCE
REVISIONS	Text	[0..1],revisions.SOURCE
ORGANIZATIONS	Number	[0..1],organizations.SHOP-DATA
CALENDARS	Number	[0..1],calendars.SHOP-DATA
RESOURCES	Number	[0..1],resources.SHOP-DATA
SKILL-DEFINITIONS	Number	[0..1],skill-definitions.SHOP-DATA
OPERATION-DEFINITIONS	Number	[0..1],setup-definitions.SHOP-DATA
MAINTENANCE-DEFINITIONS	Number	[0..1],maintenance-definitions.SHOP-DATA
LAYOUT	Number	[0..1],layout.SHOP-DATA
PARTS	Number	[0..1],parts.SHOP-DATA
BILL-OF-MATERIALS	Number	[0..1],bill-of-materials.SHOP-DATA
INVENTORY	Number	[0..1],inventory.SHOP-DATA
PROCUREMENTS	Number	[0..1],procurements.SHOP-DATA
PROCESS-PLANS	Number	[0..1],process-plans.SHOP-DATA
WORK	Number	[0..1],work.SHOP-DATA
SCHEDULES	Number	[0..1],schedules.SHOP-DATA
TIME-SHEETS	Number	[0..1],time-sheets.SHOP-DATA
REFERENCES	Number	[0..1],references.SHOP-DATA
PROBABILITY-DISTRIBUTIONS	Number	[0..1],probability-distributions.SHOP-DATA

Shop-data is the model’s very top level, and it is defined by a type, an identifier, a number, and optionally a name, a description, references, revisions, units of measurement, organizations, calendars, resources, skill definitions, setup definitions, operation definitions, maintenance definitions, a layout, parts, bills of materials, an inventory, procurements, process plans, work, schedules,

time sheets, and probability distributions. Field *INDEX* in Table 1 is a special element used by the system, its instance should be an integer, and there will be exactly one instance for a *shop-data* instance. *Type* is an attribute of *shop-data* and is an enumeration to describe types about *shop-data*. *Name* is another attribute, and there may be zero or one instance for a *shop-data* instance. *ORGANIZATIONS* is an attribute served as a pointer pointing to the table of *organizations*. (The *organizations* table view is omitted due to the page limit.) A *shop-data* instance may have zero or one instance of *ORGANIZATIONS*. *ORGANIZATIONS* in the *shop-data* table points to the field of *shop-data* in the *organizations* table.

3.2 Elements and relational tables

Elements in the XML model are used to describe fundamental features or common features of machine shop data. They are represented by tables in the database model. For illustration, two elements are selected and described here: the element *stock-level-quantities* and element *task*.

Figure 1 is another presentation view about the table/element of *stock-level-quantities*. *Stock-level-quantities* specifies the quantities of tools, fixtures, materials, or parts for various purposes, and is represented by the following attributes: *on-hand-quantity*, *allocated-quantity*, *safety-stock-quantity*, *required-quantity*, *on-order-quantity*, *back-order-quantity*, and *work-in-progress-quantity*.

stock-level-quantities
on-hand-quantity allocated-quantity safety-stock-quantity required-quantity on-order-quantity back-order-quantity work-in-process-quantity

Figure 1: Element of *stock-level-quantities*

The XML source of *stock-level-quantities* is presented as follows.

```

<stock-level-quantities>
  <on-hand-quantity />
  <allocated-quantity />
  <safety-stock-quantity />
  <required-quantity />
  <on-order-quantity />
  <back-order-quantity />
  <work-in-process-quantity />
</stock-level-quantities>

```

The element *task*, shown in Figure 2, is a child element of *tasks* and is represented by *task-definition* and *task-status*.

Task-definition specifies who the task is for (*customers*), the relative priority of this task (*priority-rating*), critical due dates for the task (*due-dates*), the process plan the task uses (*process-plan-multi-key*), the number of times the task activities need to be repeated (*repetition-count*), parameters used by the task (*parameters*), what products are required (*part-quantities*), resource requirements (*resources-required*), child work items (*child-work-items*), precedence relationships between child work items (*precedent-constraints*), and summary of estimated and actual cost data (*cost summary*). Child work items for a task may be pick orders or tool orders. *Task-status* specifies information about scheduled and actual progress towards completing the task.

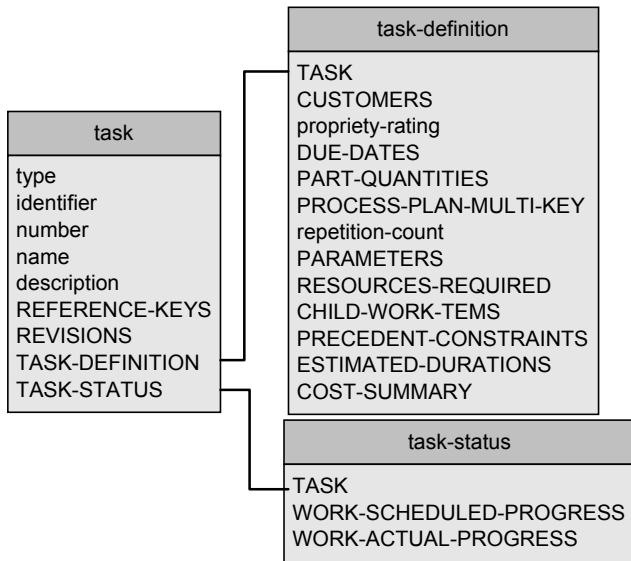


Figure 2: Element of *task*

The XML source of *task* is presented as follows:

```
<task type="" identifier="" number="">
  <name />
  <description />
  <reference-keys />
  <revisions />
  <task-definition>
    <customers />
    <priority-rating />
    <due-dates />
    <part-quantities />
    <process-plan-multi-key />
    <repetition-count />
    <parameters />
    <resources-required />
    <child-work-items />
    <precedent-constraints />
    <estimated-durations />
    <cost-summary />
  </task-definition>
  <task-status>
    <work-scheduled-progress />
    <work-actual-progress />
  </task-status>
</task>
```

3.3 Element relationships

The machine shop data model can be represented in a tree shape expression. There are different levels of relationships among elements. The element *resources* has been selected and presented in this subsection to demonstrate the element relationships. *Resources* is a collection of shop resource records and is represented by *stations*, *machines*, *cranes*, *employees*, *tool-catalog*, and *fixture-catalog* items. *Stations* specifies a collection of station-type or station-group-type resources; *machines* specifies a collection of machine-type or machine-group-type resources; *cranes* specifies a collection of crane-type resources; likewise for *employees*, *tool-catalog*, and *fixture-catalog*. Figure 3 shows the elements' parent-child relationships. These relationships are built by adding referenced or referencing element names to the database tables as table fields.

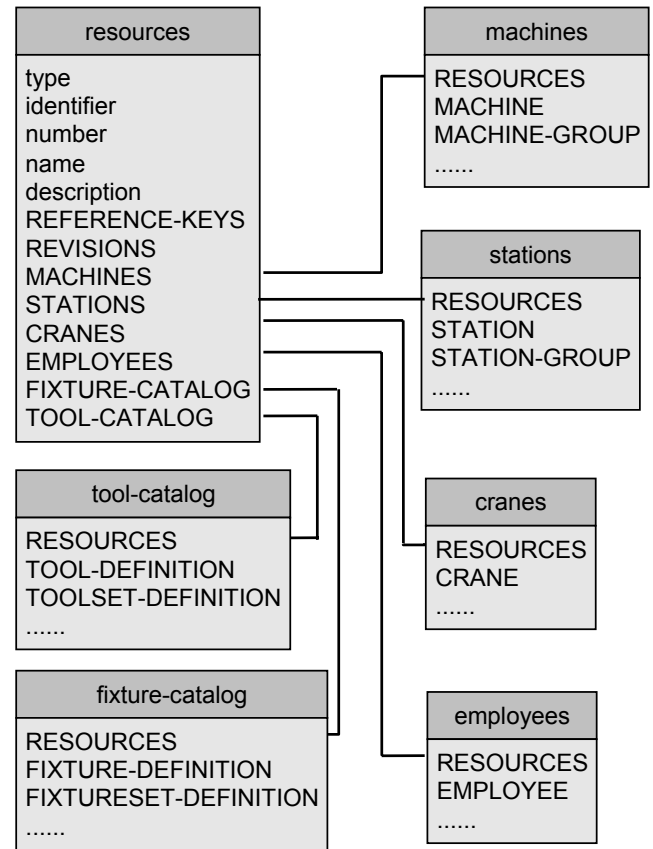


Figure 3: Element Relationships

4 SIMULATION AS A DATABASE APPLICATION

Normally a distributed manufacturing simulation (DMS) may be thought of as a manufacturing simulation that is composed of multiple software processes that are independent executing and interacting with each other

(McLean, et al. 2000). DMS needs the support from scattered data sources. A distributed database management system is used to collect, analyze, and manage data. Figure 4 shows a prototype application based on the machine shop data model. Simulation tools provide the support for design, process, manufacturing, assembly, and production management. Interfaces can be built through internal or external network.

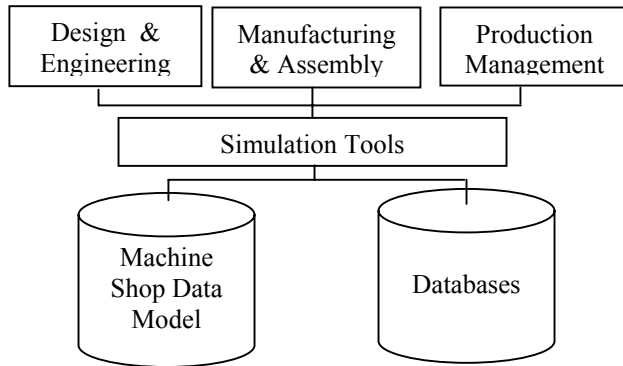


Figure 4: Implementation

5 CONCLUSIONS

This paper provides an overview of work currently undertaken at NIST to develop an information model as a neutral data format for machine shop simulations. The information model and a set of relational tables that mapped onto the information model are described. An effort to build a prototype machine shop simulator used by a pilot facility is undergoing, and the simulator can be reconfigured for use by other small machine shops. The simulator will implement the machine shop information model for the exchange or sharing of data. A simulation standards consortium, initiated by NIST, is being organized to develop pre-competitive neutral specifications for manufacturing simulation. The information model can be a candidate of a standard to be considered by the consortium or by other formal standards bodies.

DISCLAIMER

Certain commercial software and hardware products are identified in this paper. This does not imply approval or endorsement by NIST, nor does it imply that the identified products are necessarily the best available for the purpose.

ACKNOWLEDGEMENT

This project is funded in part by NIST's SIMA Program and the Software Engineering Institute's TIDE program. The authors wish to acknowledge technical contributions to this report by Fred Schenker, Project Manager of the TIDE Program, of Software Engineering Institute.

REFERENCES

- Carlisle, M., and J.E. Fowler. 2001. "Systems Integration for Manufacturing Applications Biennial Report." Fiscal Years, NISTIR 6721, National Institute of Standards and Technology, Gaithersburg, MD.
- D. Appleton Company, Inc.. 1985. "Integrated Information Support System: Information Modeling Manual, IDEF1-Extended (IDEF1X)." Wright-Patterson Air Force Base, OH.
- Inoue, H.; Y. Funyu; and S. Masami. 2000. "A New Strategic Enterprise Information Model Based on Super Parallel Evolution Computing," *Proceedings of Seventh International Conference on Parallel and Distributed Systems: Workshops*, 364–369.
- ISO 10303-1. 1994. "Part 1: Overview and Fundamental Principles." *Industrial Automation Systems and Integration-Product Data Representation and Exchange*. International Organization for Standardization, Geneva, Switzerland.
- ISO 10303-11. 1994. "Part 11: The EXPRESS Language Reference Manual." *Industrial Automation Systems and Integration-Product Data Representation and Exchange*. International Organization for Standardization, Geneva, Switzerland.
- Lee, Y.T. 1999. "Information Modeling: From Design To Implementation." *Proceedings of the Second World Manufacturing Congress* (Durham, U.K., Sept. 27-30). ICSC, Canada/Switzerland, 315-321.
- Luo, Y. 2000. "Injection Molding Product Application Activities Models." *International Journal of Advance Manufacturing Technology*. 16(4), 285-288.
- McLean, C.; F. Riddick; and S. Leong. 2000. "Architecture for Modeling and Simulation of Global Distributed Enterprises." *Proceedings of the ASIM 2000 Conference* (Berlin, Mar. 8-9). ASIM-Mitteilung Nr. AMB68. 365-374.
- McLean, C.; A. Jones; Y.T. Lee; and F. Riddick. 2002. "An Architecture for a Generic Data-Driven Machine Shop Simulator." *Proceedings of the 2002 Winter Simulation Conference*, (San Diego, CA. Dec. 8-11). IEEE Catalog Number 02CH37393C (CD).
- Object Management Group. 2003. "Unified Modeling Language." <http://www.omg.org/uml/>. UML Resource Page.
- Stoll, H.W. 1991. "Design for Manufacture: An Overview." In *Design for Manufacture: Strategies, Principles, and Techniques*. J. Corbett.; M.Doner; J. Meleka; and C. Pym (Eds.). Addison – Wesley Publishing Company, New York, NY, 107-129.
- The World Wide Web Consortium. 2000. "Extensible Markup Language (XML) 1.0 (second edition)." <http://www.w3.org/TR/REC-xml.html>.

A CLASS LIBRARY FOR MANUFACTURING SYSTEMS

Carlos Alberto Bertotto
Flávio Rech Wagner

Informatics Institute
Federal University of Rio Grande do Sul
Av. Bento Gonçalves, 9500
91501-970 Porto Alegre, Brazil
[bertotto, flavio]@inf.ufrgs.br

Carlos Eduardo Pereira

Department of Electrical Engineering
Federal University of Rio Grande do Sul
Av. Osvaldo Aranha, 103
90035-150 Porto Alegre, Brazil
cpereira@eletro.ufrgs.br

KEYWORDS

Manufacturing Systems, Simulation, Object-Oriented Paradigm, Automod.

ABSTRACT

This work presents a class library for manufacturing systems that aims at facilitating the construction of simulation models, allowing reuse and speeding up the modeling process. This library implements a modeling approach that differs from the majority of similar works in this area. It is based on the application of well known manufacturing concepts, like production routings and activities. It allows the creation of new simulations faster than other methodologies, since complex translations from the reality to simulated applications are not necessary. The development of this library was validated by modeling the production line of tractor parts. The production line case study, modeled using both Automod and the proposed class library, allowed a quantitative comparison for the validation of this work.

INTRODUCTION

The fast dissemination and widely acceptance of simulation methods contributed to the emergence of specialised environments for programming and simulation of manufacturing systems. Simulation is virtually the only methodology at present which is capable to provide accurate performance estimates for manufacturing systems design (Govindaraj et al. 1990). These environments offer tools that allow the efficient analysis from simple to complex systems. Their main limitation is the difficulty to develop new simulation models. Their model building activity usually requires specific information that is related to the internals of the adopted simulation tool. They are usually restricted to highly specialised users, since the systems usually adopt proprietary languages and particular simulation methodologies.

On the other hand, the object-oriented paradigm has taken the attention of the scientific community. Properties like encapsulation, inheritance, and reuse have contributed to make the object-oriented technology very popular for manufacturing systems simulation and deployment. The one-to-one mapping between objects in the manufacturing systems being modelled and their abstractions in the simulation model offer conditions for a better modelling. The object-oriented paradigm have been explored to allow the construction of high-fidelity simulation software for supporting the modelling and control of integrated and complex manufacturing systems.

The object-oriented paradigm and distributed objects have emerged as some of the most promising technologies since they allow a natural structure for the problem domain of industrial automation systems. This occurs because the industrial components can be easily mapped into the oriented-objects model diagrams using classes and objects. Objects encapsulate the functions of many components, modularizing the description, encouraging reuse, and allowing implementations as well in software as in hardware.

There are many commercial simulation environments for manufacturing systems. They offer many features that permit to build powerful simulations in order to develop advanced solutions for new projects through experimenting many alternative configurations. One of these simulation tools is Automod (Auto Simulations, 2002).

Automod (Auto Simulations, 1999) is a process-oriented simulator. The simulation method used in Automod is based on moving parts, representing the work in process, on resources, and on transportation entities. To create a simulation using Automod, the designer must control the traffic of parts through the system using a process specification, which indicates the correct pathway that the part must follow. This approach gives to the simulator robust and reliable features, allowing complex model constructions.

The major problem of these tools is the difficulty to build new simulation models. The modeling activity in these environments needs a detailed previous study about the tools, generally demanding highly specialised users. Moreover, these environments do not use important modeling concepts like reuse and extensions. Analysing the current state of the art, this fact is unacceptable. An integration between advantages of the commercial simulators and of the new modeling paradigms becomes mandatory. This fact may increase the quality of produced models, since developers only need to worry about the manufacturing problems, without having to worry about particularities of simulation environments.

The main goal of this work is the design of a class library for manufacturing systems that allows the development of simulation models to evaluate these systems. Basic components of these systems should be available, and it should be possible to extend them through high-level modeling features found in object-oriented systems, like aggregation and abstraction, aiming at an easier modeling of complex systems, by encouraging reuse and extensions.

STATE OF THE ART IN MANUFACTURING SYSTEMS SIMULATION

Bertotto (2001) presents a complete study of the state of the art in manufacturing systems simulation. Modeling,

control, and simulation methodologies have attracted the attention of researchers and non-governmental consortia to find new ways of developing manufacturing systems. Narayanan (1998) describes relevant efforts in the specification of object-oriented hierarchies applied to manufacturing systems simulation. Among other efforts, it is worth mentioning the application of Distributed Artificial Intelligence (DAI) and multi-agent systems, the ideas proposed by a consortium called Holonic Manufacturing Systems (HMS), and CORBAManufacturing.

Park (1997) introduces an object-oriented modeling schema for the design of automated manufacturing systems (MAS) called JR-net. To propose this schema, Park relies on the fact that the modern flexible manufacturing systems are modular and hierarchical structures, built from standard resources. This proposal is formally based on processes and resource definitions, relationships among objects, procedures to build the model, and on the model structure. Its purpose is to show that the JR-net modeling framework could be used in the same way as a graphical modeling tool for commercial simulators, such as Automod.

A CLASS LIBRARY FOR MANUFACTURING SYSTEMS SIMULATION

The main feature of this library is its modeling approach, which is different from other efforts in this area. This approach is based on activities and production routings. The production control is established by production routings that own activities and activate the respective resources. This approach makes the design easier, since the current theory of manufacturing systems is based on this kind of concepts. New simulations may be developed faster than in other methodologies, since complex translations between the reality and the simulation models are unnecessary. The class diagram of the proposed library is represented in Figure 1.

This library contains a set of classes that supports both the physical and logical modeling of a manufacturing system. The physical modeling is based on classes that represent the manufacturing system entities, like machines, robots, and tasks. The classes that represent the physical resources can be easily extended to represent other equipments that were not yet created. The logical modeling is based on classes, methods, and attributes also similar to real world logic concepts.

Class Library Specification

This library was created using UML, a standard language for object modeling containing various diagrams. The library is composed by 30 classes, linked by relationships, aggregations, and specializations. The root of the hierarchy is the `Production_Line` class. This class aggregates two other classes: `Resource` and `Master_Plan`. The `Resource` class is the root of all classes that execute some transformation on manufacturing parts. The `Master_Plan` class is the root of all classes that logically control the manufacturing system. In this class, the production plan based on client orders is generated. To create the master plan, the `Master_Plan` class uses services provided by other classes, such as `Order`, `Scheduling_Algorithm`, `Part`, `Resource` and `Daily_Capacity`. The user interacts with the model using the `Order` class, inserting orders to be produced. The `Scheduling_Algorithm` class implements the production

scheduling algorithms that will guide the master plan creation.

When the master plan is generated, the production orders should be created to control the production. Each instance of `Production_Order` will order the production of one part. If, for example, there is an order of 30 parts, there will be 30 instances of `Production_Order`. This approach makes the production flow easy. Moreover, it is necessary to update the capacity information of the involved resources. Before the creation of the production orders, `Production_Order` estimates the daily capacity of the resources, using the `Daily_Capacity` class related to the `Resource` class.

A variation of a production routing was used in the library to control the production in the various resources. The production routing is represented by an aggregation of the `Activity` class within the `Part` class. The aggregated `Activity` classes describe a logical sequence represented by a relationship that shows the next and the previous activity. An activity can only be executed if all the previous activities were executed. This aggregated structure builds a production graph that may control various activities at different levels of the production hierarchy. When an activity is completed, it activates the next instance of the `Activity` class. If there are no next activities, an answer is returned to the calling instance. This action indicates that the production of the mentioned part is concluded.

Each instance of the `Activity` class must own a default resource to control. Each activity has a relationship to the `Resource` class that will execute it. To each resource a wait queue, represented by the `Queue` class, was added. This class holds the scheduled parts if the respective resource is busy.

The `Position` class stores the physical location of the resource. The use of this class is not compulsory, but its use is interesting when it is necessary to specify the physical positions in production diagrams, shop floor lay-outs, and simulation graphic interfaces. This class could also be used to obtain the linear distance between two resources, a measure that is relevant in the computation of transportation times and capacity.

The `Resource` class is specialized in three sub-classes for modeling different types of physical entities that execute activities and move parts from one place to another: `Equipment`, `Transport`, and `Manufacturing_Cell`. With the `Manufacturing_Cell` class, the `Resource` class implements the design pattern *Composite*. This design pattern allows the composition of tree-like objects to represent part-whole hierarchies, so that users may uniformly handle both individual and composite objects (Gamma, 1995). Using this approach, the model can implement a `Manufacturing_Cell` with a complex structure, containing other cells and resources.

The `Equipment` class is specialized into sub-classes to implement different types of operations. It has a relationship with classes `Performance` and `Tool`. The goal of the first one is to store the performance attributes of the resources, so that the ideal equipment may be assigned to an activity. The function of the `Tool` class is to link a tool to a resource in order to execute a given activity.

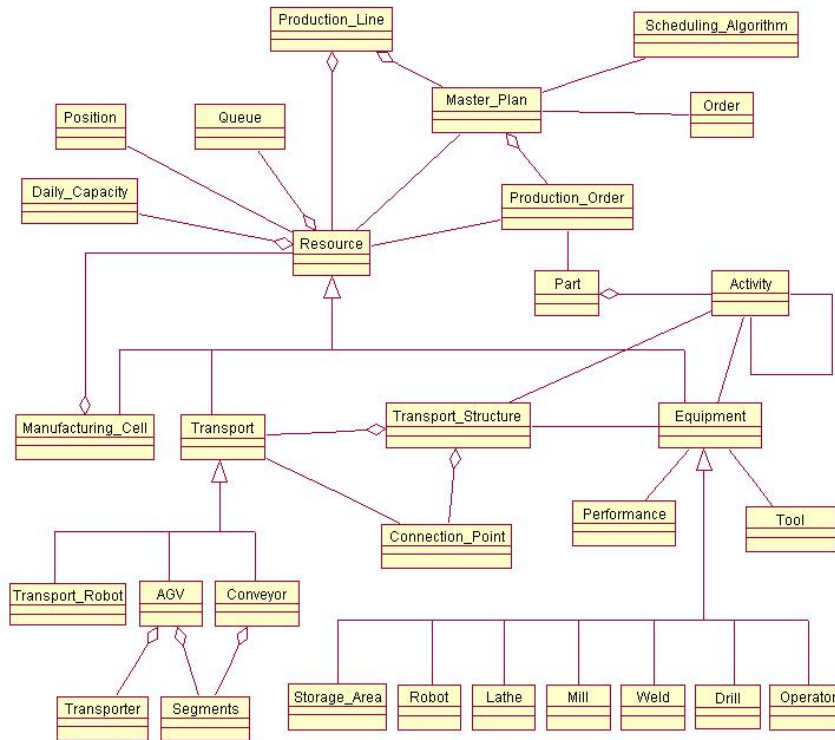


Figure 1: Class Library

Transport Infrastructure

To implement the transportation system in this library, classes were devised that allow, together with the transportation physical resources, to control and route the parts to their correct destinations. To represent the transportation physical resources, specialized classes are used: *Transport_Robot*, *AGV*, and *Conveyor*. These classes are sub-classes from *Transport* class. The representation of the routes followed by the parts is made by the *Connection_Point* and *Transport_Structure* classes.

The representation of the paths that one part can follow from a manufacturing resource to another one uses the same idea of the Network Layer of the protocol stack TCP/IP. The network layer is related to a data package transfer from the source to the destination. To arrive at its destination, the package may need many hops over routers. The routers are responsible to resolve the pathway that the package must follow (Tanenbaum, 2003).

Using the network layer concept for manufacturing systems, the transportation resources like conveyors and AGV's may represent the physical paths, as coaxial cables and optic fibers. The connection points represent the routers in the network. The manufacturing resources represent the source and destination hosts.

The *Transport_Structure* class stores all of the transport physical resources and all connection points. The transport structure gives the designer an abstract vision of the transport activities. For the *Activity* class, it does not matter which class will actually perform the transportation of a part from one resource to another. As long as the physical resources are linked by a common structure, the way how the part is conducted to another resource becomes transparent to the *Activity* class as well as to the *Equipment* class. The connection point must resolve which path the part will actually follow.

CASE STUDY: PIGOZZI'S TRACTOR AXLE PARTS MANUFACTURING CELL

The Manufacturing Cell

This case study was based on the production of tractor axle parts, as shown in Figure 2, produced by Pigozzi S/A Gears and Transmissions, located in Caxias do Sul, Brazil. This company supplies assemblies and parts for agriculture machines, off-road machines, and trucks.



Figure 2: Tractor axle part produced by Pigozzi

Seven activities may be performed by the considered manufacturing cell, as specified in Table 1.

Table 1: Manufacturing cell activities

Machine	Activity
Centering Lathe	Facing and Central Hole
Heycomat Lathe	Lather
Daewoo Puma Lathe	Lather
Brevet Gang Drilling Machine	Flange Drilling
Clever Radial Drilling Machine	Drilling
ZEWZ Curl Milling Machine	Gear Tooth Curl Milling
WMW Feather Milling Machine	Slot Feather

The information on the manufacturing cell was obtained directly from the production line. Data provided by production lay-outs, production routings, and documents have been used to create the simulation model. The main information is related to the production routing documentation. These documents contain the complete list of activities, their production times, set-up times, required tools, and other important information regarding the mentioned parts. The production routing is based on a serial process, using the most of the cell resources.

Manufacturing Cell Simulation using Automod

To model the manufacturing cell in Automod, the production routings that specify the necessary activities was evaluated. This evaluation was used with the physical lay-out to model Pigozzi's manufacturing cell. The complete graphical visualization of the model is shown in Figure 3.

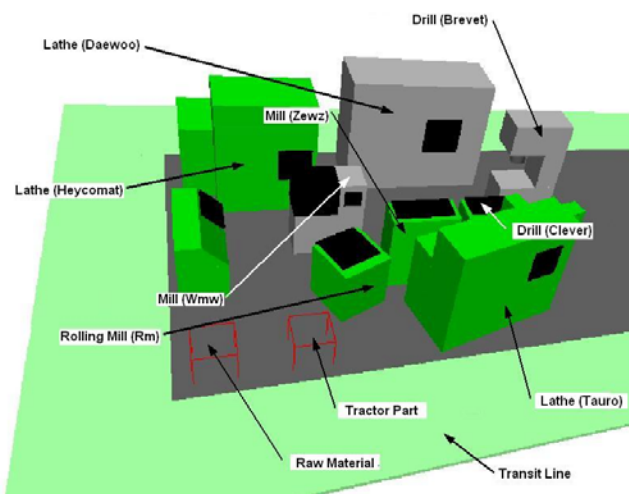


Figure 3: Automod simulation

The first step to create a model in Automod is specify the static environment where the cell is inserted. The static environment is the entities that do not move and do not execute activities over the part.

On the second step, it is necessary to place the resources on the correct local, according to the physical lay-out. These resources are the machines and equipments that execute some work over the part.

The next step is specify the entities that will transport the parts from one to another resource. In Pigozzi's manufacturing cell, there are four operators executing all the conveyance.

Afterwards the transport system creation, the physical model is completed. The next task to simulate the model is determine its functional specification. This specification is coded in a owner programming language.

There are many other activities that need to be done to produce a complete model in Automod. It follows that the development of new simulation models requires a lot of analyze, data preparation, design and simulation execution, demanding high specialized designers. Besides that, the Automod do not use important modeling concepts like reuse and extensions. Analysing the current state of the art, this fact is unacceptable. The development of each simulation spends more time and more money than any businessman expectation. The companies can not wait for simulation results to have a production line working. The machines

should be working constantly to can refund the investments done and allow the company uses all of its production capability.

Manufacturing Cell Simulation using the Proposed Class Library

This section shows the manufacturing cell simulation using the class library proposed in this work. For this purpose, classes from the library were coded in the C++ programming language. A CASE tool like Rational Rose was used to generate the C++ code of the presented classes shown on the class diagram. Some classes are not coded because they are not necessary to simulate the model. To generate the executable files was used the programming environment Microsoft Visual C++, version 6.0.

Information on the manufacturing cell lay-out and on the production routings study give the necessary input to create the model. The various physical resources, such as machines, tools, transporters, and human workers, are represented by instances of the `Resource` class. The overall control is performed by the production routing that contains all the necessary activities to produce the part.

To simulate the Pigozzi's case using the proposed library, are necessary four well defined steps that are defined in the `main` function. The first step is instantiate all the necessary classes used in simulation given to respective constructor the informations about the resource's initialisation.

The second phase is the production routing activities creation and link. This step is very important because the correct operation sequence are enclosed into the production routing.

The third step is based on relationship creation among simulation instances. This step consists of specify the correct relationships between activities and equipments. At least, the fourth stage is the customer order that determines the quantity of parts that will be produced. This order trigger the parts production.

Results

This section presents results obtained with the simulations using Automod and the presented class library, in parallel with data from the original production routing used by the company. The production lead-time to produce the tractor axle part is 3615 seconds. Both in Automod and with the class library, 20 simulation runs were executed, under the same test conditions, generating the averages shown in Figure 4.

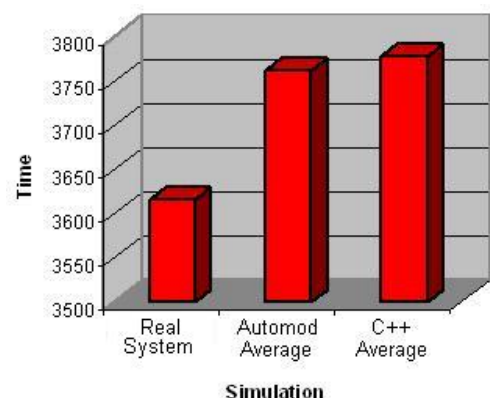


Figure 4: Simulation averages

To verify that the obtained results are reliable, some statistic measures like standard deviation, variance, and confidence interval were calculated and shown in Table 2 and Figure 5.

Table 3: Statistic measures

Measure	Automod	C++
Variance	16986	11060
Standard deviation	130	105
Confidence interval	[3505, 4016]	[3570, 3983]

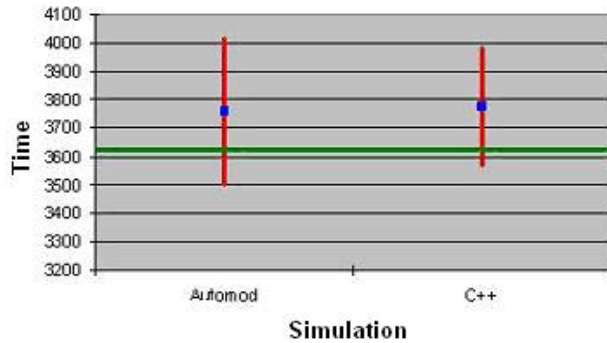


Figure 5: Confidence Interval

According to these results, the lead-time does not differ more than 5% in both simulations. The difference between the simulations and the real system is due to the fact that the system travel time is computed since the part comes into the raw material queue. If the part needs to wait some time in this queue, the travel time trends to increase.

CONCLUSIONS AND FUTURE WORK

From the case study, it is possible to conclude that the object-oriented methodology is really suitable for modeling and simulation of manufacturing systems. Real world entities could be accurately and easily represented by objects.

The class library captures the main definitions of real manufacturing systems, like production routings and activities, facilitating and speeding up the system modeling. The library uses object-oriented features, encouraging reuse and increasing productivity in the development of models.

The case study gave conclusive results about the validity of the presented library. It allowed to confirm that a fast and easy modeling of manufacturing systems is possible. By reusing classes and just defining instances, attribute values and relationships, the model is readily complete and can be simulated, generating results within few moments. The results generated by the simulations were accurate within an error of 5%, if compared to the original manufacturing cell lead-time.

The class library development started from a generic class structure, containing abstract classes that could be used in any manufacturing system. This generic set of classes was specialized and enlarged with the case study. Once the chosen application relies on manufacturing transformation processes, the final library is suitable for a very easy modeling activity.

A model was also developed with the Automod simulator. Although this model also generated accurate results, thus helping to obtain important information about the modeled reality, such as bottlenecks, Automod requires considerably more modeling time than the manufacturing

class library presented in this work. This fact is one of the main reasons for the small acceptance of similar commercial simulation environments in the market. Companies cannot wait too much for simulation results, since they need quick answers to be able to compete in the global market.

The development of classes for modeling assembly lines is a possible extension to this work. Using this idea, it is possible to create frameworks to use in specific application domains, like transformation, petrochemical, and assembly industries. Moreover, it is also necessary to develop other case studies to validate all the proposed extensions.

Other possibility is to implement the class library upon the SIMOO-RT object-oriented modeling and simulation framework (Becker, 1999). A translation tool could be integrated to SIMOO-RT, generating full models in Automod, which has better user interaction facilities. An important feature of SIMOO-RT is the automatic generation of real-time code, in languages such as AO/C++ (Pereira, 1994) or RT-Java, to control the real manufacturing plant.

REFERENCES

- Auto Simulations, Inc. *Automod User's Manual*. Bountiful, 1999
- Auto Simulations, Inc. *Automod Version 9.0*. Bountiful, 2002. 1 CD-ROM.
- Becker, Leandro Buss. *SIMOO-RT: an Integrated Object-oriented Environment for the Development of Distributed Real-Time Systems*. In: International Conference on Cad/Cam, Robotics and Factories of the Future. 15., 1999, Proceedings... [S.l.:s.n.], 1999, p. 18-20.
- Bertotto, C. A.; Pereira, C. E.; Henriques, R. V. B. *Object-Oriented Simulation and Manufacturing Systems Simulation Environments: Case Study*. In: European Simulation Multiconference, ESM, 15., 2001, Prague, Czeck Republic. Proceedings... San Diego: SCS, 2001. p. 151-153.
- Gamma, E. et al. *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading: Addison-Wesley, 1995.
- Govindaraj, T. et al. *Manufacturing Simulation Using Objects*. In: Summer Computer Simulation Conference, SCSC, 22., 1990, Alberta. Proceedings... San Diego: SCS, 1990. p.219-224.
- Narayanan, S. et al. *Research in Object-Oriented Manufacturing Simulations: An Assessment of the State of the Art*. IIE Transactions, [S.l.], v. 30, n. 9, p. 795-810, Sept. 1998.
- Park, Tea Y.; Han, Kwan H.; Choi, Byoung K. *An Object-Oriented Modelling Framework for Automated Manufacturing System*. Computer Integrated Manufacturing, [S.l.], v. 10, n. 5, p. 324-334, May 1997.
- Pereira, C. E. *Real Time Active Objects in C++/Real-Time UNIX*. In: Workshop on Languages, Compiler and Tool Support for Real-Time Systems, 1994, Orlando. Proceedings... [S.l.:ACM], 1994. 6 p.
- Pereira, C. E. et al. *Applying Multi-Agent Systems to Real-Time Industrial Automation*. In: IFAC Workshop on Multi-Agent-Systems in Production, MAS, 1., 1999, Vienna, Austria. Preprints. [S.l.: IFAC], 1999. p. 127-131.
- Rational Rose Corporation. *Rational Rose 98*. 1998. 1 CD-ROM.
- Tanenbaum, A. S. *Computer Networks*. Upper Saddle River: Prentice-Hall PTR, 2003.

The integration of Manufacturing Execution System and Product & Enterprise Resource Planning in the Logistic chain

Mounir BENAÏSSA, Abdelatif BENABDELHAFID

CERENE – University of Le Havre, France

Tel. : +33 (0) -2 32 74 46 00

mounir-ben-aissa@voila.fr; benabdelhafid@wanadoo.fr;

Adel ALIMI

REGIM - University of Sfax, National School of Engineering, Tunisia

Tel. : +216(0)-74-274-088

adel.alimi@ieee.org;

Keywords:

MES, ERP, Integration, UML and XML.

Abstract: This paper aims at the Integration Strategy problematic in order to answer the current market expectations and foresee their evolutions. It is a question of providing flexible and modular solutions making it possible to integrate the production services into the Manufacturing Execution system. So that the companies can carry out their operations with the effectiveness, quality and maximum performance and improve consequently their productivity, their reliability and their profitability. By defining the MES concept (Manufacturing Execution System), ERP (Resource Planning Enterprise) and the control System (CONTROLS). The information system is considered according to MES and CONTROLS point of view.

Our model searches to integrate the MES Concept in the Information System specially the ERP. This integration aims at exploiting the pertinent information developed by the Manufacturing Execution System. By using UML and XML we propose a definition methodology, and an exchange of technical data information.

Introduction:

The piloting of any logistic chain requires too much data-processing on the level of the physical and informational flow management. From the design beginning to the manufacture exit, a product is dealt with by various systems, most of the time distributed and developed in heterogeneous platforms. The division of this information's flow coming from these various systems is necessary to better the control on total stages of products manufacture, to be able to intervene quickly, carry out corrections, or carry out the various stages the validation. It is thus necessary to relate between the various systems taking charges of the same product manufacturing. Indeed the information division and the best of various knowledge bases, offer a better control and allow the faster resolution of problems which can occur along the various stages of the product development. This information division

allows to improve the reactivity and the company productivity. It is thus necessary to found the most effective and harmonious communication between the ERP (Enterprise Resource planning), the MES (Manufacturing Execution System) and control system or the production supervision. The communication between the various information systems from the company must evolve to a proactive relation based on the information division and the collaborative development. Today thus, the integration between the management world and that of the production execution became more than one needs; it is an obvious obligation.

We propose to integrate models of command monitoring in the logistic chain (A. Benabdelhafid and M. Benaïssa 2002). This integration supplies tools of help to the decision to the operator, which permits a piloting in real time of the production system. In the following, will be presented an idea for the integration of these models.

To better understand this environment, this document supplies the following theoretical basis: the ERP, the MES and control system definition, the context in which the MES executes its functions and the fundamental the MES interface objects. Besides this section, the second one supplies the definition of the different information systems of the logistic chain. Section three describes a methodology and technology of the information traffic between ERP, MES and CONTROLS. Section four contains the command monitoring problematic integration in the logistic chain. Section five concludes the paper with an appeal of the continuous development to the command monitoring Integration in the logistic chain.

Information system in the logistic chain

A MES occupies the information system center in the logistic chain (A. Benabdelhafid and M. Benaïssa 2002, M. Benaïssa and A. Benabdelhafid 2002). The definition can be given

as follows: a MES is a system that consists of a set of integrated software and hardware components that provide functions for managing production activities from job order launch to finished products. Using current and accurate data, a MES initiates, guides, responds to, and reports on production activities as they occur. A MES provides production activity information to other engineering and business activities in the enterprise and its supply chain via bi-directional communications. This definition is derived from a definition given in the Manufacturing Execution Systems Association (MESA) International White Paper 6 (MESA International, paper 6, 2002). In the paper, only the MES major functions in are described.

According to the MESA classification, the functions of one MES are eleven: Resource allocation and status, Dispatching production units, Data collection/acquisition, Quality management, Maintenance management, Performance analysis, Operations/detail scheduling, Document control, Labor management, Process management and Product tracking and genealogy

Manufacturing Execution System is one key element of an information system supporting a manufacturing facility. According to MESA International, there are five other key functional groupings with which a MES must interface in order to manufacture effectively a product.

The following figure (Fig.1) indicates these functional groupings and their interfaces:

These environments need to interchange with a Manufacturing Execution System (MES) environment to help manufacturing meet their needs within certain manufacturing business processes in the production area. A description of these functional groupings is below (M. Benaissa and A. Benabdelhafid 2002):

Sales and Service Management (SSM): comprises software for sales force automation, product configurations, order management, service quoting, product returns, and post-sales service.

Supply Chain Management (SCM): includes functions such as forecasting, distribution and logistics, transportation management, electronic commerce, and advanced planning systems.

Product and Process Engineering (P&PE): includes computer aided design and manufacturing (CAD/CAM), process modeling, and product data management (PDM). PDM supports information shared in a distributed environment that can cross company boundaries. The industrial enterprise has a special dependency on its product model information (product data). This product data includes the products definition that will be manufactured. CAD systems are involved in

designing a product or a facility to build a product. At this time CAD systems utilize drawings and specific documents to maintain the design information.

Controls: are usually hybrid hardware/software systems such as distributed control systems (DCS), programmable logic controllers (PLC), distributed numerical control (DNC), supervisory control and data acquisition (SCADA) systems, and other controls designed to automate the way in which the product is being manufactured.

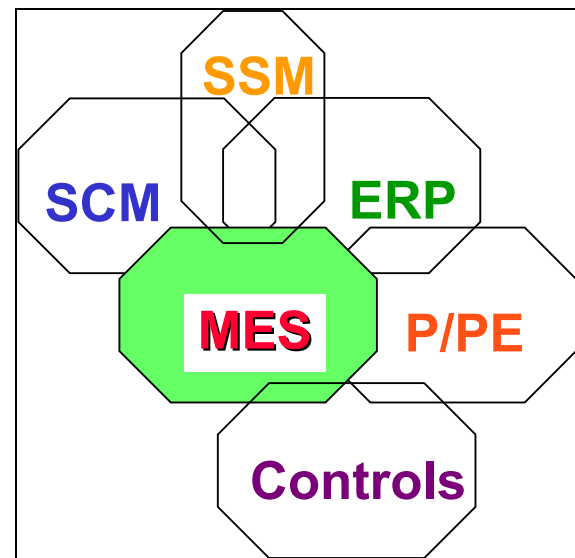


Figure 1. MES Contexte Model (M. Benaissa and A. Benabdelhafid 2002)

Enterprise Resources Planning (ERP): consists of those systems that provide financial, order management, production and materials planning, and related functions. The modern ERP systems focus on global planning, business processes and execution across the whole enterprise (intra-enterprise systems), with an accrued recent importance of aspects like supply chain planning and the whole supply chain management aspects and extending to include the whole inter-enterprise supply chain.

Information Float between ERP-MES

To better understand the data transfer between ERP-MES, the following section presents the information stream between MES and ERP.

The ERP systems are mainly concerned by the materials float management. They have transactional actions as order forms. The transactional character of this control requires the economic planning of these actions. The transaction in ERP is expressed in order of hours. The ERP modules systems can be implemented on different

platforms, probably even on systems taken away from other sites.

The command system (CONTROLS), is concerned by the management of all sorts of process parameters, as the output, the rotary speed and the temperature. The time balance is expressed in milliseconds.

The incompatibility of both worlds (ERP and CONTROLS) is caused by the transactional against the continuous character. The lack of the gap implies the capacity to reconcile process continuous data with transactional status data.

A MES ensures the link between the ERP and CONTROLS by ensuring the data acquisition, by interfacing with the command system through SCADA (Supervisor Control and Data Acquisition). The data acquisition is made in a rather direct way. Once these data collected, the MES executes a range of functions (A. Benabdelhafid and M. Benaissa 2002, M. Benaissa and A. Benabdelhafid 2002).

- It records production process data the in a data basis in real time (including the history).
- It executes the data reconciliation for the process vague data.
- It converts the process real time data into relational data of group.
- It produces all the kinds of the production reports/ratios.
- It maintains a data basis receipt.
- It manages a production program.

For the presentation of information streams of various functions(A. Benabdelhafid and M. Benaissa 2002, M. Benaissa and A. Benabdelhafid 2002), several methods are usually used (SADT, SART, MERISE, etc.). Our choice concerned the SADT (D-T.Ross, 1977) method because it offers a real «language to communicate» (IGL Technologie, 1989) furthermore it is universally known.

Integration of MES and ERP in the logistic chain

To understand the integration mechanism of the MES concept in the ERP structure in the logistic chain, we firstly present an object model describing the tasks treatment related to the received orders addressed by ERP system (Figure 2). This model is represented by Class Diagram in the format of a Unified Modeling Language UML (P.A. Muller,1997).

In Figure 3, we present a model of the work orders, maintenance and the reservation of the resources (updated of the base). The information responsibility Assumption coming from the ERP system are: Operation detailed, Manufacturing task and Work order, in order to allow the calendar update of the planned tasks. One can distinguish

production, work orders, maintenance and operation detailed.

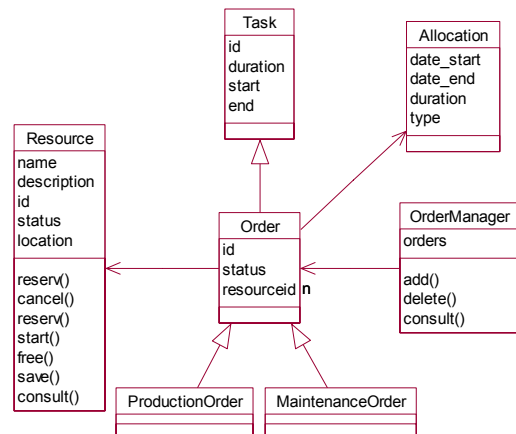


Figure 2 - Modeling of the Work Order [2]

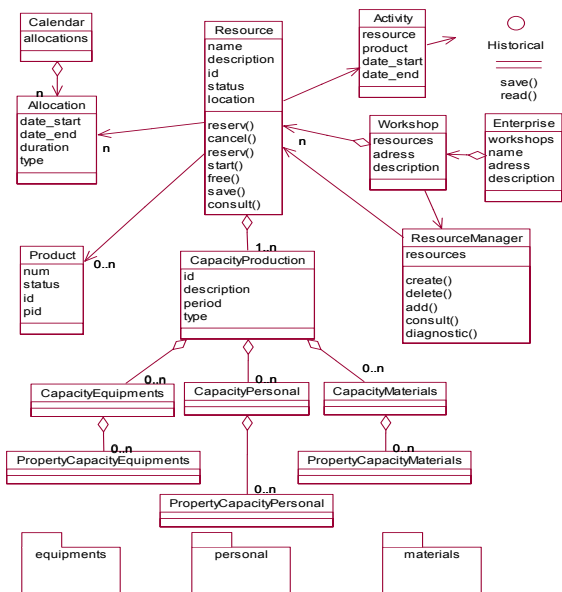


Figure 3 - Modeling of the Ressource

The model First Party (Work Order) is connected to the Ressource capacity. This model shows a bi-directional communication of the MES model and the order addressed by the ERP system.

Communication platform: MES and ERP in the logistic chain

For the exchange of information, we propose a communication platform allowing the data exchange based on messages XML and more particularly on XML/Object. This platform makes it possible to convey objects placed at the disposal by

a server and the remote use (invocation of distant methods) by a Client. This platform thus makes it possible to install a data exchange mechanism close to concept RMI (Remote Method Interface) suggested by Sun and its Java language. Whereas the Java RMI uses the serialization, i.e. the writing of an object in flow of bits thus making it possible for this object to be backed up or to be sent on the network, the XML/Object platform uses the universal format XML to describe the objects which will have to circulate. Thus, contrary to the Java RMI, which imposes the development language of the Client applications: JAVA documents XML created starting from the objects could be comprehensible by any type of client. The platform thus proposes to place at the disposal of the Client objects established on the server from which the methods could be called and used by a distant Client. This platform implements principal qualities that the existing mechanisms for the programming of applications distributed like CORBA and RMI.

The Principal advantages of this platform being:

- Total independence of the used systems environment of the.
- The mechanisms of transformation used are those showed in diagrams XML and can thus be to compile in a language different from the server, thanks to the source existing generator.

This is an example of XML Diagrams describing objects exchanged between ERP and MES system:

```
<!--schema de la classe OrdreFabrication -- !>
<element name="ordre_fabrication">
<complexType>
<sequence>
<element name="reference"
type="xsd:string"/>
<element name="quantite "
type="xsd:integer"/>
<element name="date_debut "
type="xsd:string"/>
<element name="date_fin "
type="xsd:string"/>
<element name="statut " type="xsd:string"/>
<element ref="gamme_fabrication" />
</sequence>
</complexType>
</element>
<!--schema de la classe GammeFabrication -- !>
<element name="gamme_fabrication">
<complexType>
<sequence>
<element name="reference_produit"
type="xsd:string"/>
<element name="description"
type="xsd:string"/>
<element ref="operation_detaillee"
minOccurs="1" />
</sequence>
</complexType>
</element>
```

```
<!--schema de la classe OperationDetaillee -- !>
<element name="operation_detaillee">
<complexType>
<sequence>
<element name="reference" type="xsd:string"/>
<element name="date_debut"
type="xsd:string"/>
<element name="date_fin" type="xsd:string"/>
<element name="marge_amont"
type="xsd:string"/>
<element name="marge_aval"
type="xsd:string"/>
<element name="designation"
type="xsd:string"/>
<element name="num_ressource"
type="xsd:string"/>
<element name="priorite" type="xsd:string"/>
</sequence>
</complexType>
</element>
```

Conclusion

This type of integration allows a good industrial processes control and allows the exploitation of the new technology contributions in term software package to integrate, several improvements are possible such as the use of the Data Mining tools for the Performance Analyses .

Reference

A. Benabdelhafid and M. Benaissa : “The integration of the supervision in the MES environment within the Integrated Logistics Information System (ILIS)” The 2nd IEEE International Symposium on Signal Processing and Information Technology, Marrakech 2002.

M. Benaissa and A. Benabdelhafid “ The integration of the supervision in the MES environment within the framework of the Extended Enterprise” (14 TH European Simulation Symposium Exhibition, ESS 2002,Dresden, Germany).October 2002.

D-T.Ross “Structured Analyses: A language for communication Ideas”. IEEE Transaction on software Engineering, vol3, page 16-34,1977.

IGL Technologie « ADT, un langage pour communiquer » Eyrolles édition, 1989.

A. Benabdelhafid, M. Benaissa, “Manufacturing Execution System in the Integrated Logistics Information System”, SCS , Londre 2002

MESA International, “ ES Explained: A High Level Vision,” White Paper 6 (Pittsburgh: Manufacturing Execution Systems Assoc., 1997).

P.A. Muller, «Modélisation Objet avec UML», EYROLLES, 1997.

MODELLING AND SIMULATION OF A PILOT SCALE MILK PASTEURISATION UNIT

Ben Griffin* and Tom O'Mahony§
Advanced Control Group
Department of Electronic Engineering
Cork Institute of Technology
Cork, Ireland
*e-mail: bgriffin@cit.ie, §tomahony@cit.ie

KEYWORDS

First principles modelling, plate heat exchanger, heat transfer, high temperature short time pasteurisation

ABSTRACT

This paper is concerned with the modelling of a high temperature short time pasteurisation plant. A first principles model of the plate heat exchanger based on the energy balance equations of the system is developed and its transient and steady state response is obtained using the finite difference method. The accuracy of a low-order model derived from the response of the high order finite difference model is also examined and was seen to give a similar response. Finally, a black-box model is obtained from closed-loop process input-output data using MATLAB's System Identification Toolbox and its performance is compared with the first principles model. It is anticipated that the developed models will prove beneficial in the analysis and design of advanced control techniques that will improve the performance of the process.

Nomenclature

A	heat transfer area, m^2	<i>Subscripts</i>
c	specific heat capacity, J/kgK	i inlet
D_e	equivalent diameter, m	o outlet
h	heat transfer coefficient, W/m^2K	h hot fluid
k	thermal conductivity, W/mK	c cold fluid
l	thickness of plate, m	s stainless steel
\dot{m}	fluid mass flow rate, kg/s	f fluid
Nu	Nusselt number (dimensionless)	p plate
Pr	Prandtl number (dimensionless)	
Q	heat transfer rate, W	
Re	Reynolds number (dimensionless)	
T	temperature, $^{\circ}C$	
U	overall heat transfer coefficient, W/m^2K	
V	Volume, m^3	
ρ	density, kg/m^3	
μ	viscosity, Ns/m^2	

INTRODUCTION

Pasteurisation is a very common process in the food industry. It consists of heating the product, in this case milk, to a certain minimum temperature for a certain minimum time, with the intention of removing pathogenic micro-organisms. There are various pasteurisation methods

but the High Temperature Short Time (HTST) (Alfa-Laval 1986) method is the most common due to its simple adaptation to continuous processing. Plate heat exchangers (PHE) (Alfa-Laval 1986) are used extensively in this process to raise the temperature of the product to the required pasteurisation temperature. They are also widely used in other areas of the food, chemical and pharmaceutical industries. Therefore, the modelling of plate heat exchangers is vital if advanced control or statistical process monitoring is to be used to improve the performance and reliability of these processes.

Steady state models of heat exchangers are easily constructed and are used when planning heat exchanger networks. Dynamic models of heat exchangers are more difficult to deduce since the liquid temperature within the heat exchanger is both a function of time and position within the heat exchanger. In (Khadir et al. 2000) a low order two-input two-output model is used to model each section of the plate heat exchanger of a milk pasteurisation plant. The model's order was intentionally kept low as it was to be used in a model predictive control strategy. However, after some necessary modifications to the model and filling in the systems parameters (volumes, flow rates, heat transfer coefficients etc.) the results obtained were very poor, requiring ad-hoc modifications of the time constants and gains to obtain reasonable results. Consequently, model confidence is reduced, as there is little correlation between the model and the actual physical parameters of the system. The first principles model developed in this paper differs from the proceeding in that each section of the heat exchanger is segmented to create a lumped parameter model by applying the finite difference method. The results obtained using this model were very accurate.

A difficulty with first principles modelling is that it requires expertise and may be difficult and time consuming to develop. Also, even if a first principle model is developed, the physical properties may be very difficult or impossible to obtain from the process. Therefore, in addition to the first-principles modelling, this paper will present results arising from black-box modelling methodology (Ibarrola et al. 1998; Negiz et al. 1998). The accuracy of the black-box model will be compared vis-à-vis the first-principles model in terms of their ability to predict the plant response, from a given input sequence.

The developed models can then be used to optimise the existing PID controller and to investigate the potential performance improvements arising from the use of an

advanced control strategy such as feedforward, cascade, gain scheduling or model predictive control.

DESCRIPTION OF PLANT AND PASTEURISATION PROCESS

The process modelled in this paper is a pilot scale high temperature short time pasteurisation plant in Moorepark Technology Ltd. at the Dairy Products Research Centre, Fermoy, Cork. The plate heat exchanger (Tetra Plex C3 with a total of 58 plates) is comprised of three sections: heating, regeneration and cooling, see Figure 1. The regeneration section uses the hot pasteurised milk to heat the incoming raw milk, which at the same time cools the pasteurised milk, thus yielding increased energy efficiency in both heating and cooling cycles.

The pasteurisation process is described as follows; with reference to Figure 1, the raw unpasteurised milk at a temperature of 2-6°C is pumped at 500l/h from the balance tank into the regeneration section of the PHE. This milk is preheated to about 63°C by the returning pasteurised milk. The preheated milk then goes into the heating section where it is heated to the pasteurisation temperature, say 74°C, using hot water as the medium. The milk is held at the pasteurisation temperature for 15 to 20 seconds in the holding tube. After the holding tube the milk enters the regeneration section where it is cooled by the incoming raw milk to about 14°C. Finally, the milk enters the cooling section where it is cooled to 2-3°C using chilled water as the medium. The hot water used in the heating section is heated in a brazed heat exchanger (BHE) by regulating the flow of steam through the BHE.

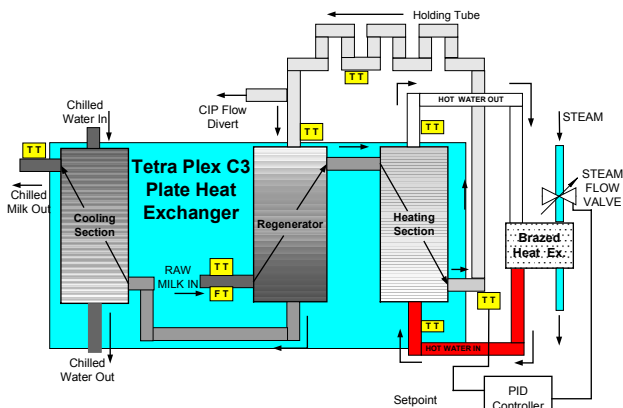


Figure 1: Block Diagram of Pasteurisation Unit

The legal requirements for HTST pasteurisation varies from country to country but, in general, it is required that the milk be heated to 72°C for 15 seconds. This is a point on the time-and-temperature lethal effect curve for the Tubercle Bacilli micro-organism (Alfa-Laval 1986). However, a safety factor is built into the holding time to allow for varying velocities of milk through the pipes. Furthermore, since perfect control is never possible in the presence of disturbances, a safety factor is also built into the pasteurisation setpoint. The setpoint is set to a value greater than 72°C, its value dependent on the maximum expected variance in the pasteurised milk temperature.

Disturbances that can influence the output milk temperature are, for example, steam pressure variations, pump speed variations and raw milk fat content and temperature. The potential exists here to use more advanced control techniques than the PI controller, typically used in these situations, to save energy and avoid over heating of the milk. In this case the variance would be reduced by taking the disturbances into account, and thus the setpoint could be reduced from say 75°C to 73.5°C, saving approximately 5700J of energy per litre of milk pasteurised.

The plant is monitored and controlled by ABB's Sattline distributed industrial control system. The temperature from seven temperature transmitters (TT) and the raw milk flow rate from a flow transmitter (FT) are logged, the location of the transmitters are shown in Figure 1.

FIRST PRINCIPLES MODEL

In (Khadir et al. 2000) first-principle models of the heating and regeneration sections of plate heat exchanger were developed. Each section of the PHE was considered to be a single plate separating the product and medium, as shown in Figure 2(a). The temperature gradient inside the heat exchanger was not considered and each PHE section was modelled as a two-input, two-output system as shown, in transfer function form, in Figure 2 (b), where G_{nm} is a first order system. As stated previously, very poor results were obtained due to the over simplification of the system.

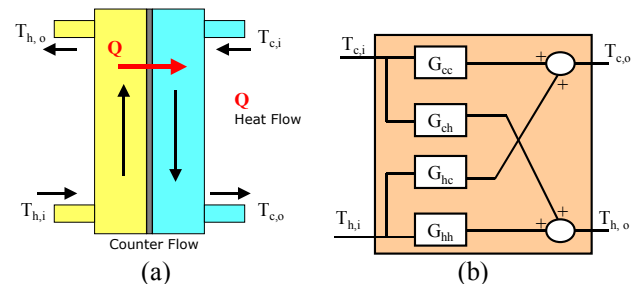


Figure 2: (a) One Section of the PHE, (b) Simple Model of one Section of the PHE

If a more accurate first principles model of the heat exchanger is required then the temperature gradient along the length of the heat exchanger has to be considered, Figure 3 (a). The thermal network modelling technique can be used to achieve this. This technique uses finite difference methods to convert distributed parameter systems into standard state variable form. Such a technique is outlined for a shell and tube heat exchanger in (White 1991). Using this method, the heating section of the plate heat exchanger, for example, is divided up into a number of nodes, each representing a finite volume, see Figure 3(b). An energy balance is then applied around each node as follows:

(rate of change of *energy storage* in node) = (net rate of *energy gain* from fluid flow into and out of the node) + (net rate of *energy gain* from convection with the plate metal)

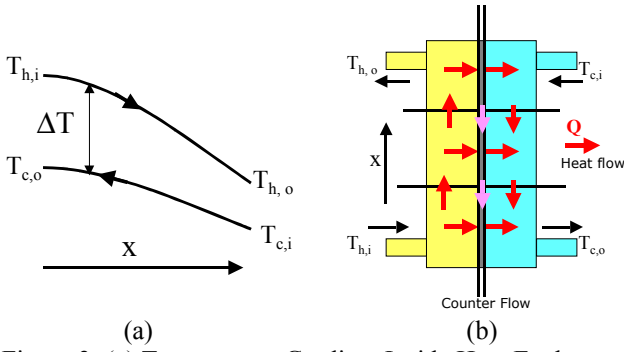


Figure 3: (a) Temperature Gradient Inside Heat Exchanger, (b) Finite Difference (Lumped Parameter) Method

For example, the energy balance for the j^{th} node on the hot fluid side is given by

$$(\rho_h c_h V_h)_j \frac{dT_{h,j}}{dt} = \dot{m}_h c_h (T_{h,j-1} - T_{h,j}) + h_h A_{h,j} (T_{p,j} - T_{h,j})$$

A similar energy balance is applied to the cold fluid side and an energy balance is also applied to each of the plates nodes. Thus, if the heating section of the PHE is divided into N sections, i.e. N hot fluid nodes, N cold fluid nodes and N plate nodes, this will yield a system of $N \times 3$ differential equations, with two inputs and two outputs (the fluids inlet and outlet temperatures) which can be solved numerically.

The next step in developing the first principle model is the determination of the various physical properties of the system. The specific heat capacity, density and thermal conductivity of water and stainless steel can be found in (Incropera and Dewitt 1996), for various temperatures. The dimensional properties of the heat plate exchanger were obtained from physical measurements and the information obtained from the manufacturers data sheets, such as the plate's surface area. The flow pattern inside the plate heat exchanger is very important since the heat transfer coefficient depends on the fluid velocity and the fluid velocity will depend on how many parallel channel flows there are in each section of the PHE.

The convection heat transfer coefficient, h , between the channel fluid and the plate wall can be found using convection correlations. A heat transfer correlation for turbulent flow in a gasketed PHE is given by (Schlunder 1983)

$$Nu = 0.2 Re^{0.67} Pr^{0.4} (\mu / \mu_{wall})^{0.1} \quad (1)$$

for $Re \geq 1000$. Knowing the Nusselt number, Nu , the heat transfer coefficients h_c and h_h can then be calculated using

$$h = Nu.k_f/D_e$$

where D_e is the equivalent diameter of the channel and k_f is the fluids thermal conductivity. The overall heat transfer coefficient, U , between the two fluids on either side of the plate can be evaluated using the equation

$$\frac{1}{U} = \frac{1}{h_c} + \frac{l}{k_p} + \frac{1}{h_h} \quad (2)$$

where l is the thickness of the plate and k_p is the thermal conductivity of the plate material.

To assess the accuracy of the convection correlation in Eq. (1) the following equation can be used to determine U if the steady state inlet and outlet temperatures are known

$$Q = UA\Delta T_{LM} \quad (3)$$

where ΔT_{LM} is the log mean temperature difference, given by

$$\Delta T_{LM} = \frac{(T_{h,i} - T_{c,o}) - (T_{h,o} - T_{c,i})}{\ln[(T_{h,i} - T_{c,o}) / (T_{h,o} - T_{c,i})]}$$

for a counter flow heat exchanger. The total heat transfer rate, Q , can be obtained using the following equation

$$Q = \dot{m}_c c_c (T_{c,o} - T_{c,i}) = \dot{m}_h c_h (T_{h,i} - T_{h,o}) \quad (4)$$

However, in an industrial scenario not all the inlet and outlet temperatures of each section of a PHE will be recorded. In many cases, as in this case, the output of one section is the input to the next section, both internal to the PHE. Therefore finding ΔT_{LM} is not always possible. However for the pasteurisation plant in question, ways around this problem were found.

In analysing the heating section, it was noticed that, a flow divert valve at the end of the holding tube (used in the CIP cycle) can divert the hot pasteurised product to the drain or balance tank, instead of passing it through the regeneration section. Thus, in steady state, the input product temperature, $T_{c,i}$, of the heating section is approximately equal to the raw product input temperature of the regeneration section. Now that the two inlet and two outlet temperatures of the heating section are known, this enables ΔT_{LM} to be calculated and hence U for the heating section.

In the regeneration section neither of the outlet temperatures are recorded and are internal to the PHE. However, in the regeneration section the product is flowing on both sides of the plates at the same mass flow rate, implying that $\dot{m}_c c_c = \dot{m}_h c_h$, therefore Eq. (4) becomes $(T_{c,o} - T_{c,i}) = (T_{h,i} - T_{h,o})$ and rearranging yields:

$$(T_{h,o} - T_{c,i}) = (T_{h,i} - T_{c,o}) = \Delta T$$

i.e. a constant temperature difference across the section. In this case Eq. (3) becomes

$$Q = UA\Delta T \quad (5)$$

Therefore, in this case if three of the four inlet and outlet temperatures are known then U can be calculated using Eq. (5). If the cold medium flow in the cooling section is turned off then the steady state outlet temperature of the regeneration section, $T_{h,o}$, is approximately equal to the outlet temperature of the cooling section. Thus, $T_{c,i}$, $T_{h,i}$ and $T_{h,o}$ are known and the overall heat transfer coefficient, U , for the regeneration section can be calculated.

The theoretical values of U calculated using Eq. (2) for both sections of the PHE were very close to the values of U calculated from the steady state analysis of the PHE outlined above. Thus the convection correlation in Eq. (1) is very accurate for this PHE.

The flow rate of the product is monitored and logged. However, the flow rate for the hot water, used as the heating medium in the heating section, was unknown. If the inlet and outlet temperatures are known then Eq. (4) can be used in steady state conditions to evaluate \dot{m}_h . In this case

the product flow rate = 495l/h and the medium flow rate in the heating section was calculated to be approximately 920 l/h.

With knowledge of the product and medium flow rates, the volume of the product, medium and stainless steel in the PHE section, the heat transfer area and the individual heat transfer coefficients, the various thermal resistances and capacitance for each node can then be computed. Since the system matrices for the $N * 3$ state system were generated using software, a large value, $N = 50$, was used. Obviously the higher N is, the greater the accuracy of the results. This was particularly true for the regeneration section where the temperature drop is much greater than that in the heating section.

To examine the steady state accuracy of the finite difference state variable models, Figures 4 and 5 illustrate the model's steady state temperature variation along the length of the regeneration and heating sections respectively. The output temperatures are in good agreement with the temperatures measured in steady state conditions from the real plant. Note, as mentioned previously, the temperature difference along the regeneration section is constant. However, in the heating section $\dot{m}_c c_c \neq \dot{m}_h c_h$ and so the temperature profile of figure 5 is obtained.

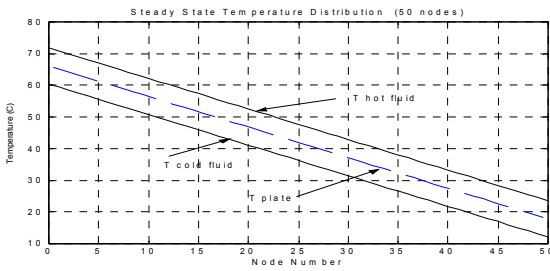


Figure 4 : Steady State Temperature Gradient Inside Regeneration Section

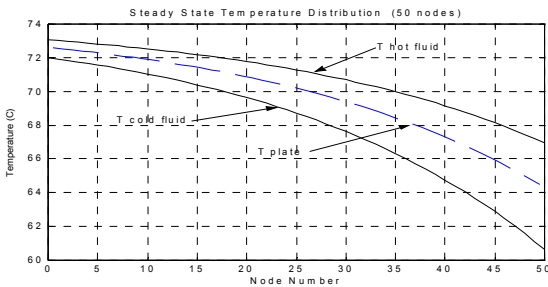


Figure 5: Steady State Temperature Gradient Inside Heating Section

Figures 6 and 7 show the transient response obtained for the regeneration and heating sections respectively. These figures illustrate the change in the outlet temperatures to a step change of 1°C, applied to the inlet temperatures. Note that a change of 1°C in the cold fluid input temperature $T_{c,i}$ has a minimal effect on the cold fluid outlet temperature, especially for the heating section.

An examination of these transient responses would suggest that the system may be modelled by a set of first order and first order with time delay models, having the structure of Figure 2(b). These models were obtained by

estimating the models coefficients (time constant, gain and delay) from the transient response produced by the finite difference model. The responses of these low order models are also shown in Figure 6 and 7.

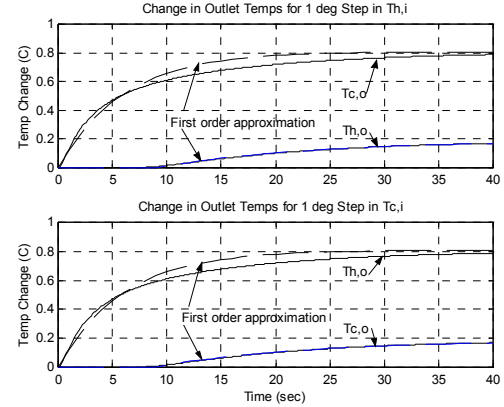


Figure 6. Transient Temperatures in Regeneration Section

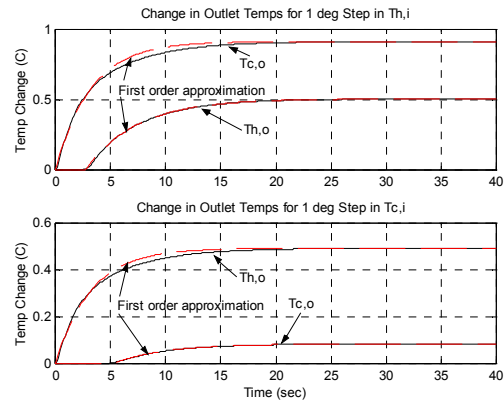


Figure 7. Transient Temperatures in Heating Section

To simulate the behaviour of the milk pasteurisation unit the regeneration and heating sections were coupled together in The Mathworks simulation tool Simulink. $T_{c,o}$ of the regeneration section is connected to the $T_{c,i}$ of the heating section. $T_{c,o}$ of the heating section is fed back to $T_{h,i}$ of the regeneration section through a delay block and a first order model with a gain of 0.98, which represents the holding time and the temperature drop across the holding tube. To assess the accuracy of the model, real plant data (the plant was under closed-loop PI control) was applied as inputs to model and the outlet temperatures of both the model and plant compared. The 'raw milk' (in this case water) temperature data was applied to $T_{c,i}$ of the regeneration section and the hot water temperature out of the brazed heat exchanger is applied to $T_{h,i}$ of the heating section. Figure 8 shows the plant and model $T_{c,o}$ and $T_{h,o}$ temperatures of the heating section. The model response is very close to the actual values again indicating the heat transfer coefficients calculated using Eq. (4) are very accurate initial estimates. The Mean Absolute Error, MAE, for $T_{c,o}$ was 0.14°C and 0.51°C for $T_{h,o}$. The values of h_c and h_h may be varied slightly to yield a more accurate result. The hot water flow rate may also be varied slightly since this was an estimated value as previously explained. For example, increasing h_c from 6706W/m²K to 7150W/m²K reduce the MAE of $T_{c,o}$ to 0.09°C.

Using the four first order (plus dead time) transfer functions, that approximate the finite difference models transient responses in Figures 6 and 7, resulted in a value of MAE equal to 0.11°C for $T_{c,o}$ with $h_c = 7150\text{W/m}^2\text{K}$ which is a marginal increase over that for the high order finite element model (0.09°C).

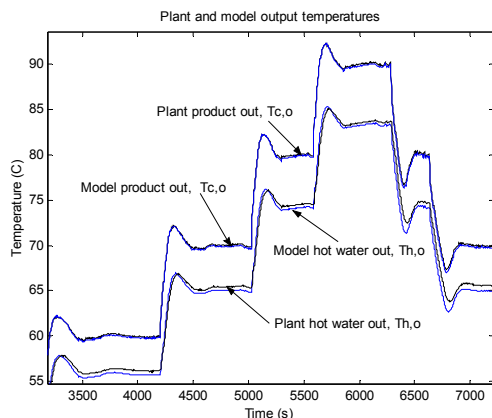


Figure 8: Plant and Model Output Temperatures

BLACK-BOX MODEL ESTIMATION

A number of contributions have covered the black-box modelling of a pasteurisation plant. In (Ibarrola et al. 1998) a laboratory scale pasteurisation unit is modelled, where each section of the PHE had the same structure as that in Figure 2 (b), where G_{nn} was a first order plus dead-time system. The parameters of the models were estimated using an output error structured algorithm. In (Negiz et al. 1998) high order ARX time series models were found for the different plant variables in respect to the plant variables that had a correlated affect on them. Each of these papers used open-loop process input-output data to estimate their model parameters.

Since two inlet and two outlet temperatures of each section of the PHE are not recorded it is not possible to apply the method in (Ibarrola et al. 1998). Instead a two-input two-output black-box model is estimated from closed-loop process data. The input signals were the raw milk temperature and the hot water from the BHE and the outputs were the hot water out of the PHE and the pasteurised milk temperature. The System Identification Toolbox from MATLAB (Ljung 1991) was used to identify a suitable model. An accurate eight state, state space model was identified from data for a particular run and resulted in a MAE of 0.04°C for $T_{c,o}$ and 0.08°C for $T_{h,o}$ for that run. However, when data from a different run was applied, the output temperatures were not as accurate, MAE = 0.1°C for $T_{c,o}$ and 0.25°C for $T_{h,o}$. A possible reason for this is the 'raw milk' in temperature varied very little in the data set used to identify the model, therefore the full influence of the 'raw milk' inlet temperature may not be fully characterised by the model. Furthermore, this single model will only give accurate results for a particular product flow rate and if conditions change then a new model will have to be identified, since the parameters of the model cannot be modified knowingly. To overcome this problem, (Ibarrola et al. 1998) calculates numerous model parameters for different flow rates. In (Negiz et al. 1998) the flow rate was

considered as one of the inputs to the ARX models developed.

FURTHER WORK

Further work involves developing models for the brazed heat exchanger and modelling the characteristics of the steam valve so that the relationship between the control variable (valve percentage open) and the steam flow rate can be obtained. The complete model can then be used to assist in tuning of the existing PI controller and the development and evaluation of more advance control techniques.

CONCLUSIONS

The models developed using the finite difference method proved to be very accurate in predicting the process output. Furthermore, slight modifications of the physical properties of the PHE can lead to more accurate results if conditions have changed, or physical properties were under or over estimated.

A low order two-input two-output model comprised of first-order and first-order plus dead-time models derived from the transient response of the high order finite difference model was shown to be very accurate also, with little difference between the two models. This low order model may be useful if model predictive control strategies are to be implemented.

In the case of the black box model, accurate models can be obtained for specific flow / holding times etc. However if one of these variables changes a new model will have to be identified or a model that takes these parameters as inputs needs to be developed.

REFERENCES

- Alfa-Laval. 1986. *Dairy Handbook*, Alfa-Laval, Lund, Sweden
- Ibarrola, J.J., J.C. Guillen, J.M. Sandoval and M. Garcia-Sanz. 1998. "Modelling of a high temperature short time pasteurization process", *Food Control*, Vol 9, No. 5, pp 267-277
- Incropera, F. P., D.P. Dewitt. 1996. *Introduction to heat Transfer*, John Wiley and sons, Inc, New York
- Khadir, M. T., J. Richalet, J. Ringwood, B. O'Connor. 2000. "Modelling and predictive control of a milk pasteurisation in a plate heat exchanger", *Proc. Foodsim*, Nante, France pp216-220
- Ljung, L, 1991, *System identification toolbox*, The MathWorks Inc
- Negiz, A, P Ramanaukas, A Cinar, J. E. Schlessor and D. J. Armstrong. 1998. "Modeling, monitoring and control strategies for high temperature short time pasteurization systems – 1. Empirical model development", *Food Control*, Vol 9, No 1, pp 1-15.
- Schlunder, E. U. (1983). *HEDH – Heat exchanger design handbook* (Vols 1-3). Washington, USA: International Centre for Heat and Mass Transfer
- White, J. R., Course material on System Dynamics, Chemical and Nuclear Engineering Dept. University of Massachusetts Lowell, available from www.profrwhite.com

Acknowledgements: The Authors wish to acknowledge the support of Enterprise Ireland under the Food Factory of the Future Program.

HUMAN PERFORMANCE MODELLING WITHIN ASSEMBLY LINE SIMULATIONS

Tim Baines, Linda Hadfield, Steve Mason, John M Kay
Cranfield University
Cranfield
Bedford MK43 0AL

John Ladbrook
Ford Motor Company
Dunton
Essex

KEYWORDS:

Human Performance Modelling, Manufacturing Systems Design, Discrete Event Simulation, Decision Tools

ABSTRACT

This paper is part of a wider project to extend the scope of Discrete Event Simulation (DES) of manufacturing systems, by developing tools which can take into account the impact on system performance of direct workers (Human Performance Modelling, or HPM). In order to develop such tools, it is first necessary to demonstrate the existence of variability in worker performance, and define its magnitude, profile and impact on the overall system. The paper describes a three month empirical study which was carried out on an assembly line. Movements of components through a number of operations on the line were monitored, and additional contextual data were collected concurrently. The results of the study show that the performance of direct workers varies considerably. This finding has implications for the effectiveness of both DES and work study, and indicates the importance of the development of Human Performance Modelling tools and methods.

INTRODUCTION

Modern demand-led approaches to manufacturing, such as JIT (just-in-time) and lean manufacturing impose a requirement on manufacturing systems to be flexible and able to respond rapidly to changes in demand for end products. This presents a challenge for the designers of manufacturing systems, who are expected to be able to respond rapidly to changes in requirements, and produce the optimum design for a manufacturing system in the shortest possible time. Systems designers use Discrete Event Simulation (DES) to respond to this challenge, designing and re-designing the required systems through the use of computer models, which can be easily adapted and re-evaluated without the need to invest time and effort in changing the physical systems. However, the results predicted by DES models have consistently been shown to be different from the results which occur in practice.

Observation of the results predicted by DES models of systems involving workers against wholly automated systems suggests that the basis for this difference lies in the inability of DES to represent the performance of direct workers accurately. Consequently, in order to improve the effectiveness of manufacturing systems, there is a need for the development of techniques which can be used to extend the scope of DES by taking into account the performance of direct workers (Baines and Kay 2002).

The work described in this paper is concerned with the development of such tools and techniques for modelling the performance of direct workers, known as Human Performance Modelling (HPM). A three month study was carried out to track the performance of direct workers on an assembly line. Data were collected on the processing times taken by each worker to complete a task. The study was based on two null hypotheses: that the times taken to complete a task by different workers would be equivalent; and that the time taken to complete different tasks would be equivalent. However, the data revealed that considerable variations exist both between workers and between tasks. The study produced distributions of task completion times which vary both in the form of the distribution and the mean values observed. These empirical distributions will be used in the development of a modelling tool to improve the accuracy of DES.

The structure of the paper is as follows: a discussion of the background to the use of assembly lines in manufacturing, and the assumptions of worker behaviour embodied by this; a description of the design and execution of the empirical study; a summary of the main results from the analysis of the empirical study; and a discussion of the implications for the development of HPM tools, and the design of manufacturing systems in general.

BACKGROUND

The origins of the modern assembly line lie in the development of the "Scientific Management" approach to industrial production, originally designed by FW Taylor in the early 20th century, and put to practical application by

Henry Ford in his factory at Highland Park, Detroit, Michigan, constructed in 1913 (Womack et al. 1990). The basis of scientific management involved two aspects: the principle of interchangeability and the division of labour (Askin et al. 1993).

The principle of interchangeability refers to the standardization of the individual components that make up a finished product, such that components are interchangeable between product units. Division of labour refers to the breaking down of a job into smaller component tasks and activities which can be easily learnt with a minimum of training of the workers. In this way, workers, as well as parts, become interchangeable units, able to repeat the actions required of them in a standardized, controllable and highly predictable manner.

In order to coordinate the overall production process, the various tasks which are carried out sequentially must be equivalent in the time taken to execute them. If this is not the case, bottlenecks and part shortages will occur as the product unit moves down the line. The tasks are carefully designed, and predicted execution times (work standard times) calculated based on observation (work study) of the times typically taken to carry out similar activities. In this way, designers attempt to ensure that the line is 'balanced', i.e. that different tasks can be completed in equivalent time, and that the line can therefore run smoothly.

The design of the assembly line thus implies two assumptions. Firstly, because of the way in which tasks are designed to be as simple and standardized as possible, the performance of different individual workers should be equivalent. Secondly, in order for the line to be balanced correctly, the time taken to perform different tasks must also be equivalent. There should therefore be no scope for individual human variability, which in theory has been designed out of the system.

Discrete Event Simulation models, which are designed to allow system designers to investigate and predict the performance of manufacturing systems, have therefore to date treated direct workers as simple components, and assumed no variation between them (Bernhardt and Schilling 1997). However, the predictions produced by DES models of highly automated systems are notably more accurate than those involving direct workers (Baines and Kay 2002). The existence of this 'gap' suggests that individual workers can in fact have an impact on overall system performance.

This gap between manufacturing system performance as predicted by DES models, and the outputs which such systems generate in practice, creates problems for system designers. It would therefore be helpful for system designers if the scope of DES could be extended by developing tools to simulate worker behaviour (Human Performance Modelling, or HPM). This paper is part of a wider project aimed at developing HPM tools.

EXPERIMENTAL DESIGN

The aim of the study was to investigate the true nature of human variability in performance on an assembly line. In order to do this, some mechanism was required to allow the collection of information on worker performance.

Design engineers assess the effectiveness of manufacturing systems on the basis of many performance measures generated by DES models. Typical measures include: output of the system (e.g. units per hour), lead time and quality (e.g. viable parts per million). Such measures provide an insight into the overall performance of a proposed system. Micro measures can also be taken of individual machines, such as utilisation etc.

A number of performance variables relating to the performance of individual workers were considered in the study:

- Work station activity time: The time taken to complete a single task.
- Error rate: An indication of how well an operator conducts a task. Errors are any deviation from product specification, which may result in unserviceable product ("scrap") or product which requires additional processing to become serviceable ("rework").
- Dependability: Given that all conditions for a task to commence are met, when does the operator start the activity in response to a request?

In addition, a set of indirect performance measures are also necessary to reflect worker wellbeing. These do not have an immediate impact on individual task performance, but may feedback into the overall performance of the manufacturing system in the long term. Therefore, absenteeism, accident rate and staff turnover were also chosen as measures.

This paper will concentrate on work station activity rate. Various methods of measuring activity rate were considered. Methods involving direct observation were discarded for two major reasons: firstly, because of objections from the workers taking part in the study, and the concerns that direct observation would potentially affect the performance which was being measured, as occurred in the classic Hawthorne studies (Mayo 1997). Secondly, direct observation was infeasible because of the impracticality of observing workers through three 8 hour shifts a day, 5 days a week.

The assembly line used in the study is divided into ten main work zones, which were included in the study, with additional testing and finishing zones, which were not included because the processes performed on them are atypical. Product units travel along the line on platens, passing through all of the zones in turn. Each of the zones included in the study contains a number of tasks ("operations"), which are performed by a team of workers, varying in size from 6 to 12 people. Each individual on the team works on all the operations in his/her zone, moving on to the next operation on the hour. Because

concerns were voiced by the workers about the monitoring of individual performance, it was decided to monitor one operation on each of the ten zones, the operations chosen being those in each zone with the highest planned work station activity time. This allowed the researchers to monitor differences in activity time on different operations and for different teams of workers, but without being able to relate specific performance back to identifiable individuals.

The sequence of events for each operation on the line is as follows:

- Operator presses button to move new platen to work station.
- Platen arrives at work station and comes to a stop.
- Operator performs required activities on product unit at work station.
- Operator presses button to release platen currently at station, and move new platen to work station.

In order to monitor the amount of time spent working on each operation, switches were installed underneath the work stations to record platen movements. Using this method, it was possible to monitor the work stations 24 hours a day, 5 days a week, over a 12 week period, generating approximately 150,000 data points for each operation monitored.

The plant operates a twenty four hour, three times eight hours shift system, with three crews of workers. Of the three crews of workers working on the line during the study period, two were directly comparable, but the third consisted of a smaller number of workers per team, some of whom were required to perform more than one task. Although data were collected for this crew (Crew B), they were not included in the analysis.

In addition to the workstation data, data were also collected on a number of contextual factors (eg breakdowns), other performance measures (accident rate, absenteeism, staff turnover) and a number of potential contributory factors, relating to the individual attitudes of the workers, the physical environment and the organizational environment.

DATA ANALYSIS

The data initially collected consisted of the time (down to the nearest eighth of a second) at which the switch on the underside of the platen was activated, and the position of the switch (1 or 0, where 0 indicated platen arriving at station, and 1 platen leaving station). This allowed the calculation of activity time (time at which platen was stationary at station) and the gap during which the platen travelled between two stations (or, more accurately, during which the next platen in the sequence travelled to the

station). The overall cycle time was calculated as the sum of work station activity time and the subsequent gap.

The data were examined to remove those data points which corresponded to a scheduled break, or the handover time between two crews of workers, at the end of a shift. The remaining data were plotted as frequency histograms of activity times and cycle times for each of the 10 operations monitored.

The initial histograms showed extreme workstation times of several minutes in some cases, as well as extremely short workstation times (less than 5 seconds). These short workstation times were clearly too short to possibly reflect actual work taking place, and it is to be assumed that they reflect cases where the worker moves the platen on without carrying out any work on the product unit, either because the product unit is faulty due to a problem upstream, or because the worker has been “working ahead”, ie working on the product unit before it arrives at the station.

Various smoothing methods were tried to remove these extreme workstation times, including discarding the top 5% of workstation times for each hour (ninety-fifth percentile method) and discarding any jobs for which the overall cycle time (workstation time + subsequent gap) exceeded 27 seconds (line speed plus 20%). These smoothing methods removed the most extreme data points, but analysis showed that they did not significantly affect the shapes of the resultant distributions.

RESULTS

As described above, the principles on which assembly lines are based assume that assembly tasks are designed to be as similar as possible in terms of the time required for their execution. Only if this is the case can the line be balanced and hence run smoothly.

The most striking finding from the present study, however, was the marked differences in the distributions of activity time generated by the different tasks. Figures 1 and 2 show comparative frequency distributions for two of the operations monitored. Neither of the distributions has a statistically normal (bell-shaped) distribution. Figure 1 shows a bi-modal distribution, i.e. in addition to a large peak at an activity time of 8 seconds, there is also a smaller peak at 14 seconds. This suggests that there exists at least two different ways of completing the task, although the exact meaning of this cannot be interpreted from the numeric data alone, and direct observation is required to investigate this further.

Figure 2 shows an operation which has a single dominant peak at 19 seconds, but also a large number of extremely short activity times of 5 seconds or less. It is postulated that this is caused by the operator working “up the line” on product units which have yet to arrive at the station proper, and then pressing the button repeatedly to move the completed units through to the next work station.

However, as before, this behaviour can only be confirmed through direct observation.

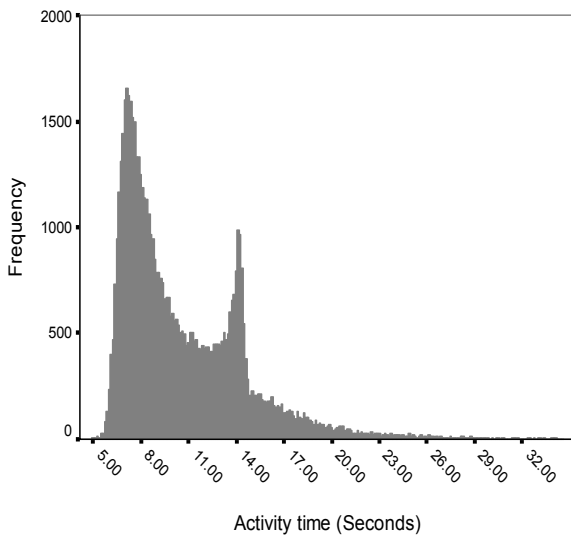


Figure 1: Activity time frequency histogram for Operation A

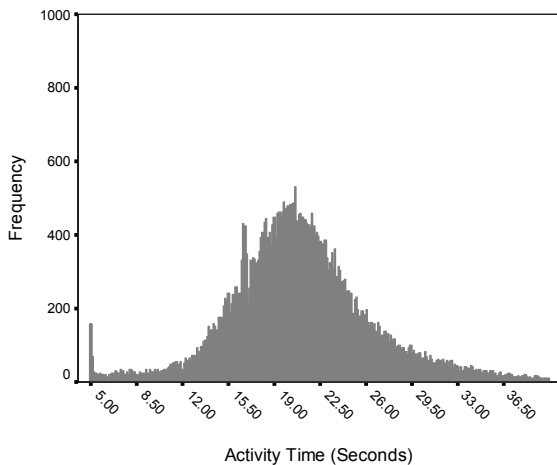


Figure 2: Activity time frequency histogram for Operation B

In addition to the assumption that all tasks are substantially equivalent in the time they take to complete, similarly it is thought that different operators will take the same amount of time to complete a task. Time series analysis of the data has shown that there may be significant differences in both the mean value and spread of activity time from hour to hour, as individual workers in the team rotate around the tasks in their work zone. Figure 3 shows data taken from a typical eight hour shift, where the differences from hour to hour can clearly be seen.

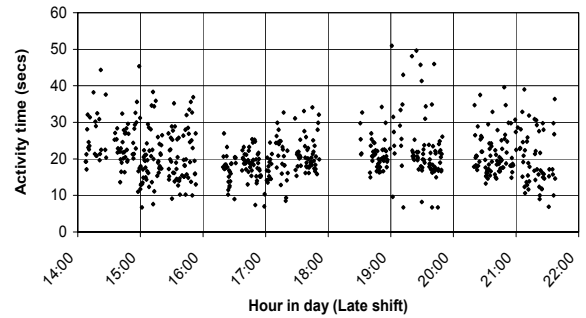


Figure 3: Time series scatter plot of activity time variations over shift

CONCLUSIONS AND FURTHER WORK

The results of the study show that the performance of direct workers, as measured in terms of process times, varies considerably. This implies that current methods of predicting system performance, which ignore the effect of worker behaviour, are inadequate to explain system behaviour fully. Contrary to the hypothesis that the times taken to process these operations would be consistent, the study demonstrated that process times vary widely. It would seem, therefore, that the assumption that worker performance can be easily predicted is incorrect. This finding has implications for the effectiveness of both DES and work study, and indicates the importance of the development of Human Performance Modelling tools and methods.

A first step in the exploitation of these findings to extend the scope of Discrete Event Simulation will make direct use of the distributions of activity times. Work is currently underway on the development of a computer modelling tool which will allow the user to select from a number of distributions of activity times and incorporate these into models of the assembly line. This offers a significant improvement on present methods, in which activity times are assumed to be uniform, or randomly distributed.

A further development of the work concerns the additional data which has been collected with relation to contextual factors, individual worker attitudes, environmental conditions and organizational factors (Hadfield et al, 2002). This information will be used to investigate a theoretical framework of the factors which affect direct worker performance. The identification of the most significant factors will permit the generation of numerical models ('Micro models') of the relationships between input factors and performance measures (Mason et al. 2002). These will be used in the development of more sophisticated HPM tools, which will allow system designers to investigate the impact of varying the underlying causal factors. It will also lead to a more robust understanding of the factors which affect the performance of direct workers.

Through improving designers' ability to predict the behaviour of direct workers, HPM offers the potential of designing more accurate models, improving the flexibility and effectiveness of the resulting systems.

REFERENCES

- Askin, Ronald G & Standridge, Charles R, *Modelling and Analysis of Manufacturing Systems*, John Wiley & Sons, Inc, New York, 1993
- Baines, T & Kay, J M, "Human performance modelling as an aid in the process of manufacturing system design: a pilot study", *International Journal of Production Research*, Vol 40(10), 2002, pp2321-2334
- Bernhardt, W and Schilling, A., "Simulation of Group Work Processes in Manufacturing", In *Proceedings of the 1997 Winter Simulation Conference, Atlanta, Georgia*
- Hadfield, L, Fletcher, S, Mason, J P, Baines, T and Ladbroke, J, "A Theoretical Framework for Human Performance Modelling", *Proceedings of the Second International Conference on Systems Thinking in Management (ICSTM02)*, Salford University, 3rd-5th April 2002
- Mason, S, Siebers, P, & Baines, T, "Humans: The Missing Link in Simulation?", *Proceedings of the OR Society Simulation Study Group Two day Workshop, University of Birmingham, 20-21st March 2002*
- Mayo, E, "Hawthorne and the Western Electric Company", In *Organization Theory: Selected Readings*, D.S. Pugh (ed), Fourth Edition, Penguin Books Ltd, London, 1997, Chapter 17, pp355-368
- Womack, J P, Jones, D T, & Roos, D, *The machine that changed the world*, Maxwell Macmillan International, Oxford, 1990

VISUAL SIMULATION IN MANUFACTURING MANAGEMENT: SOME OBSERVATIONS

Peter Lawrence
School of Business Systems
Box 63B, Monash University
Victoria 3800 AUSTRALIA
Email: peter.lawrence@infotech.monash.edu.au

ABSTRACT

Although computer simulation has been around as an operations management tool for several decades the increasing availability and power of animated simulation modelling tools has added a new dimension to modern simulation studies. This paper presents a brief discussion of anecdotal evidence, drawn from several recent Australian case studies, to support the view that animation, particularly accurate 3-dimensional modelling, has added greatly to the utility and acceptance of simulation in Australian industry. These case studies, all conducted over recent years, include an automotive production line, a proposal for new plant, an airport baggage handling system and robot work cell design.

INTRODUCTION

Simulation, and particularly Discrete Event Simulation (DES), have been around as operations management techniques in the manufacturing industry for nearly fifty years (Kelton, Sadowski and Sadowski 1998:13, Robinson 1994:3). The basic principal of building a computer model that tracks the changes in a system's state with time has changed little over this period. Many special purpose languages have been developed to aid in the design, execution and analysis of such models including commercial languages such as GPSS, SIMAN, SLAM and SIMSCRIPT.

Many modern simulation languages have changed little in structural terms since their early inception. Typically, simulation models were built in text editors and then executed from a command line prompt to produce a text based result file of primarily statistical output. This statistical data forms the basis of the analysis and provides a means to both describe and assess the performance of the system and also build confidence in the validity of the simulation model being used. However, this form of results presentation is very technical in nature and is hence likely to make diminishing degrees of sense as it is passed from the simulation analyst to the industrial engineer to the operations manager and finally the company board.

This difficulty in the understanding of the implications of simulation results as the client becomes more remote from the technical details of the company's operations, can undermine management confidence in the results and thus

limit the potential of simulation as a tool to improve operations. This is particularly true for small businesses where the cost of employing the necessary in-house technical expertise may be prohibitive.

However, developments in computer hardware and software have had one obvious impact on the functionality of specialist simulation languages in that there has been a substantial advance in the quality of tools available to allow visualisation of the system. In particular, the past decade has brought significant improvements in the quality and utility of 2-dimensional, and more recently 3-dimensional, animated representations of the physical system being modelled (Kelton *et al.* 1998:14, Bell, Anderson, Staples and Elder 1999).

It can be convincingly argued that the addition of animation, though it does little or nothing to improve the numerical accuracy of the simulation and its resultant system analysis, has nevertheless added a new dimension to the usefulness of such models in industrial application. There is evidence to suggest that simulation study clients such as industrial engineers and operation managers see simulation animation as a value added component of the traditional study (Bell *et al.* 1999, Kirkpatrick and Bell 1989, van Dijk, Jobing, Warren, Seeley and Macri 1996).

Similarly, this paper will argue that animation allows non-technical managers to more readily appreciate the results of operational simulation models and also facilitates their participation in the process of model development and analysis. In this way simulation, though still requiring expertise to safely employ, is no longer warily viewed as a black box by management and can more readily become part of their palette of decision making tools.

Also, while discrete event simulation is seen as primarily a tool for analysing, and hence solving complex transaction processing problems, this paper argues that it also has a significant educational role to play in industry and that this role is substantially enhanced by the inclusion of animation in the study.

Evidence to support these assertions is presented by way of the accounts of outcomes and client comments from several commercial consultancy projects for various Australian manufacturing companies.

CASE STUDIES

All of the case studies presented here involved the construction of simulation models using the specialist simulation language systems known as QUEST and IGRIP produced by Delmia (formerly Deneb Robotics). QUEST is designed for the construction of discrete event simulations that model the overall flow of resources through and between work cells and is thus well suited to looking at performance of production lines. IGRIP, on the other hand, is best suited to more detailed kinematic and dynamic models and so is most useful in investigating the operational details within individual work cells.

Both of these tools present a detailed and dimensionally accurate 3-dimensional animation of the system as part of their output.

Case 1: Visual Simulation in Car Manufacture

With the introduction of a new model an automotive plant must generally undergo significant re-tooling. In this case, the addition of various new features and options on

upcoming new models meant that some significant changes needed to be made to the operation of the final stages of the production line. The simulation modeled the final stages of the production process. Vehicles, upon reaching the end of the assembly line are driven onto, or into the limited queue space for, one of the wheel and headlamp alignment stations and then, after completion of the alignment process onto the test rolls. At the test rolls a bank of tests are performed on each vehicle at speeds of up to 120 km per hour to ensure that the vehicle is performing to specification.. Once again, should all of the test rolls be occupied vehicles need to wait in the limited queue space between the alignment rigs and the test rolls.

The production line had been operating smoothly for some years but considerable uncertainty was about to be introduced due to the introduction, for the first time, of Independent Rear Suspension (IRS) to certain models. This necessitated the installation of a new, separate, laser wheel alignment station thus adding the following complications to the system:

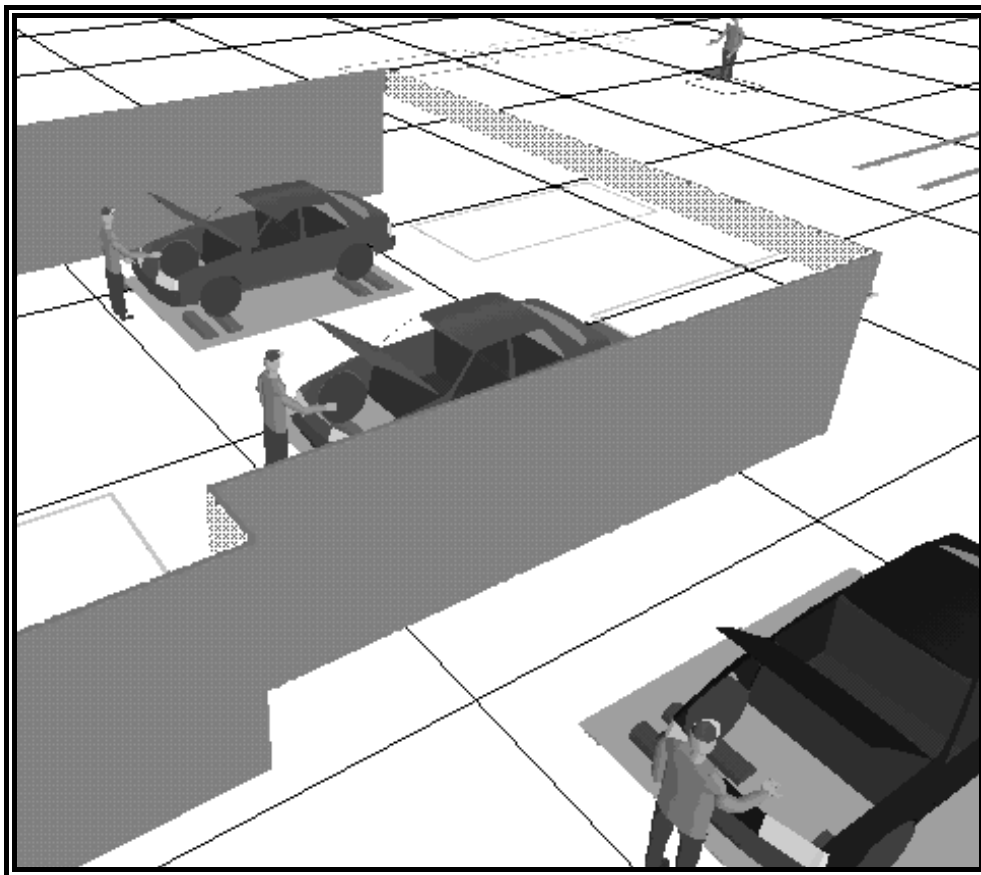


Figure 1: Simulation of Automotive Production Line

- The addition of this third rig while being an extra alignment resource was the only station capable of aligning IRS vehicles and so complicated the process of assigning vehicles to alignment stations as they came off the production line.
- The time to align IRS wheels on the new rig was three times that of traditional vehicle suspension and hence was a potential resource bottleneck.
- The test rolls traffic patterns, and hence the necessary traffic flow rules for operators would be altered unpredictably by the new alignment process and hence test roll cycles times may need to be revised to ensure smooth operation.
- The limited space for vehicle queues meant that there was little scope for accommodating even transient bottlenecks because once the wheel alignment queues filled the production line would need to be stopped with lost retail production value of around \$40,000 per minute.

A simulation model (a still picture taken from the output of which is presented in figure 1) was used to study the operation of the wheel alignment, wheel balancing, headlamp alignment, test rolls, final inspection, detailing and water testing of vehicles. Major objectives included:

- determination of target cycle times for these processes,
- determination of preferred location for construction of new balance rolls,
- investigation and identification of potential traffic congestion problems,
- confirmation that existing labour levels would be sufficient,
- testing of proposed rules for selecting which aligner and test rolls each vehicle should use.

The model was successfully used to determine workable traffic flow rules and determine target values for station cycles times and IRS vehicle build rules (the minimum number of non-IRS vehicles that must roll off the production line between two IRS vehicles). However, it transpired that the study had additional benefits to the client.

An obvious advantage of animation in this case study occurred in the model validation process. In building a simulation model it is necessary to make decisions about the scope and level of the model. The modeller must judiciously disregard the parts of the system (scope) and the details (level) that will have zero or negligible impact on the objectives of the study. However, in doing this assumptions are being made about the system's performance and so there is always the chance that such assumptions, and hence the model, are invalid. For this reason model validation is a vital (though often neglected) part of the process of conducting a simulation study.

A sometime useful technique for model validation is the Turing test where model output data sets are mixed with real system data and an expert such as the operations manager is given the task of determining which are which

(Robinson 1994:144). The inclusion of animation, and the ability to control its playback speed so that an entire shift could be viewed in real-time or in a few minutes, allowed a variation on the Turing test. Plant engineers were able to watch the animated operation and judge whether or not it was operating with the same parameters as the real plant. However, using animation rather than numeric output meant that shop floor operators and not just graduate Industrial Engineers were able to contribute to this process.

The most obvious educational advantage of this study came about during the model-building phase. At this point the analyst needs to get a detailed understanding of how the production line operates so that an accurate model can be built. A significant part of this process involves interviewing the relevant engineers and asking a series of detailed questions. What became apparent in this process was that there were many operational details about which the engineers were unsure or where their views were not self-consistent. This often came as a revelation to the engineers concerned and led to many a visit to the line to study just how it did operate. Hence the process of simply trying to build a model had positive educational outcomes for the engineers.

Animation meant that factory floor staff such as supervisors and leading hands could be involved in the design and decision making process. The proposed new system was presented to them via the animation and several suggestions and improvements to the operating rules of the line were made. These workers, normally excluded from operational decision making, were able to make useful contributions and hence the company benefited from a knowledge base not often enough tapped.

In fact, this process of involving production staff in the model building exercise further served as education for the engineers in that it allowed production workers to watch the plant operating as the engineers believed it did and correct their misconceptions. It turned out, for example, that the rules given by the engineers for how drivers choose which of the test roll stations to move each vehicle onto were inefficient, even unworkable, and that shop floor workers had long ago changed the rules. Without the animated model the engineers would never have learned about their misconception.

Comments from various plant managers also made it quite clear that they would not have been willing to trust, or even commission, this study had they not been able to visualise the system and its operation. In short, were it not for the animation capability simulation would not have been used.

Case 2: New Plant Layout

The purchase of new plant is very often a major expenditure and decisions such as these cannot be made lightly. Simulation without animation has served as a useful tool in assessing the operation of proposed new plant in the past but, at the end of the day, it can hope to do little more than convince the engineers. The decision makers at board level are often faced with approving a multi-million

dollar capital expenditure whose operation is largely a mystery and of which they have little or no visual conception.

Visualisation via traditional 2-dimensional engineering drawings is difficult enough for experienced engineers but is nearly impossible for most general managers. This is sometimes overcome by building a physical scale model of the system but such models give little or no insight into plant operation and are generally difficult to amend as the project develops.

Animated simulation models were built for several new plant projects including a high volume fastener manufacturing plant, a proposed gas bottle refilling plant (see figure 2) and several domestic and international air terminal baggage handling systems. The final layout of the bottle filling plant was reached through many iterations and varied significantly from the original conception. The animation included every pallet and every bottle in the plant and so, by watching the plant's operation, engineers were able to ensure that their operating procedures were viable and more easily identify potential bottlenecks.

However, while the engineers developing the proposal readily admitted the value of such a simulation in their design process they expressed even greater enthusiasm for the animation as a tool for their final presentation to the board. The team pointed out that, despite their working for many months on the project, approval for such expenditure in industry was often granted or denied by senior managers based upon a brief written executive summary and a 15 minute presentation to the board. They felt that the ability to give their board a vivid dynamic image of exactly what they would be getting for their money would significantly enhance their chances of gaining support.

A similar story relates to the construction of major airport baggage handling systems, a detail from one of which is shown in figure 3. While the simulation study was able to prove that the proposed system would meet throughput requirements this was never really in doubt. The primary purpose was to produce a video presentation of the system, complete with explanatory voice over, to present to senior managers for final approval.

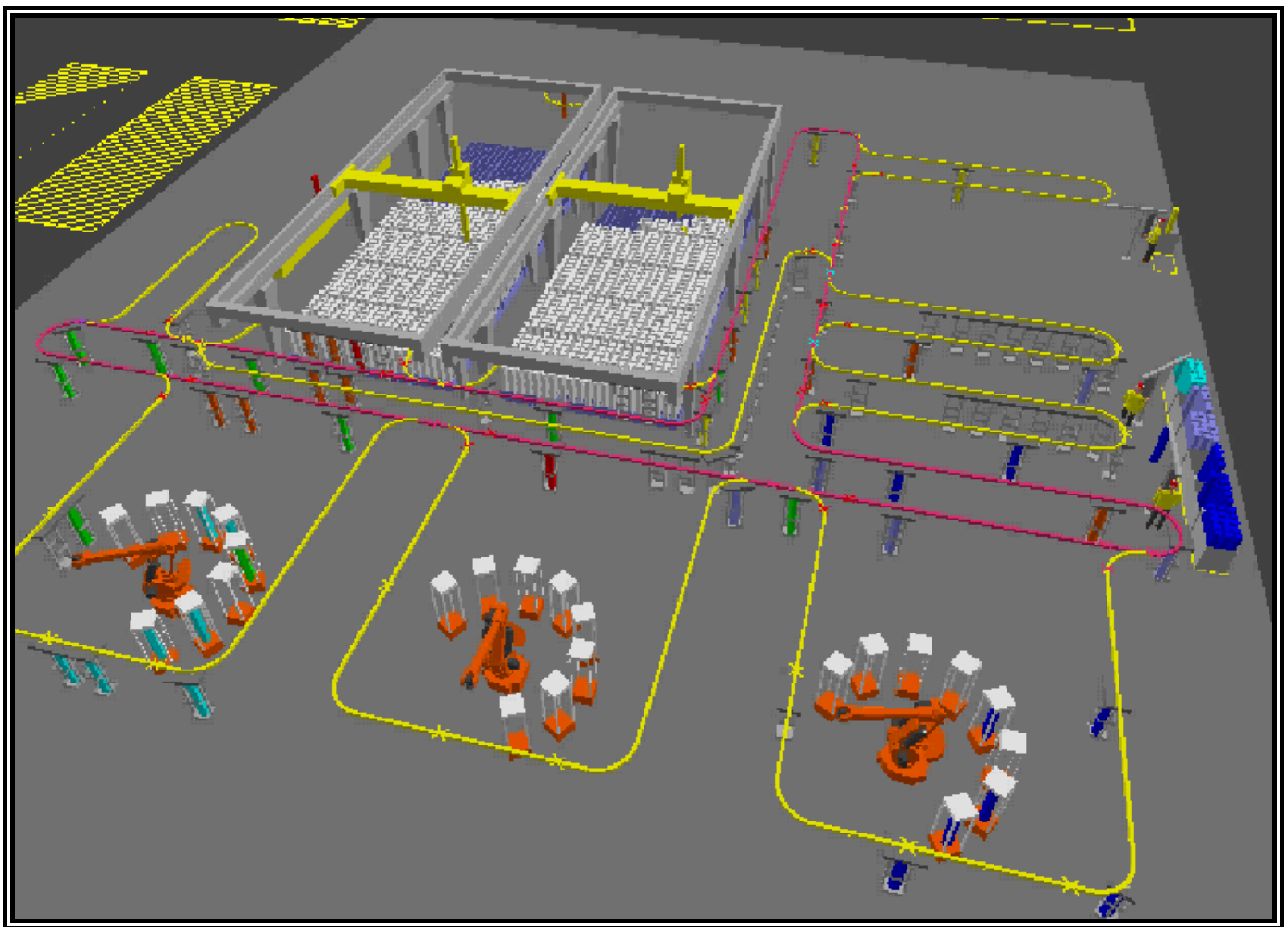


Figure 2 Green Field Gas Bottling Plant

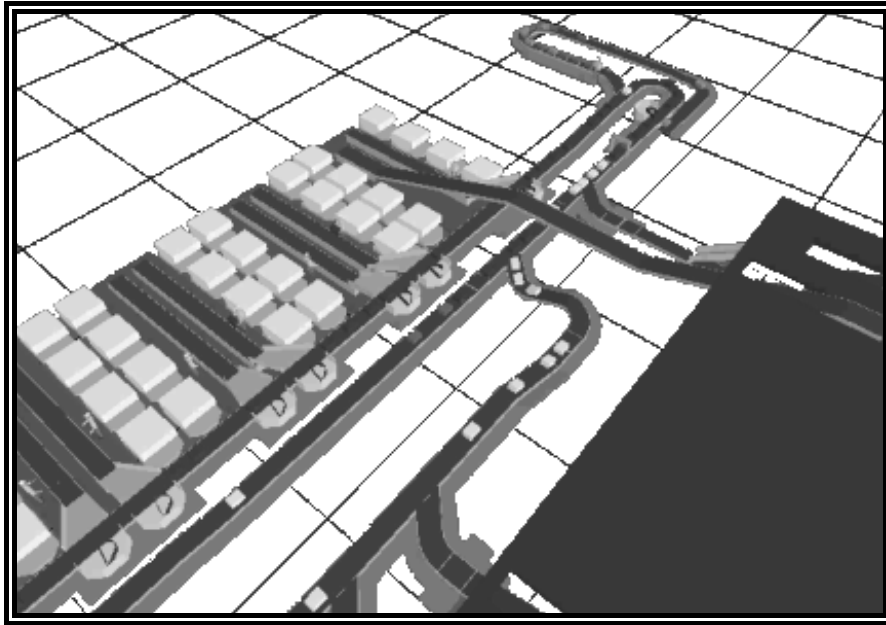


Figure 3 Proposed Baggage Handling System

Case 3: Spatial Arrangement in Robot Work Cells

A large number of consulting jobs involved the design of work cells, particularly those involving industrial robots. While inverse kinematic simulations of such systems can be used to measure robot performance and cycles time, or even automatically generate the robot's program code, in very many cases what clients were most interested in was the visualisation of the work cell and establishing the optimum spatial relationships between components. More often than not, no robot code was generated in these studies but screen shots of the animation were sent to engineering workshops and used as the basis for design and construction of the final cell.

In fact, 3-dimensional animated simulations of this nature are now used by one of Australia's leading robot vendors as a standard marketing tool. Animated models of work cells are routinely developed as part of the tendering process for new clients and are also used as "attention getters" at trade shows. The senior sales and management staff within the company have repeatedly expressed the view that such animations give them a significant competitive advantage.

CONCLUSIONS

The conclusions presented here can at this stage be considered as only preliminary since none of the claims presented for the utility and broad industrial acceptance of detailed animation in Australian simulation studies is as yet supported by great rigour. There remains much scope to more methodically survey industry to support the assertions made here. It can nevertheless be said that repeated anecdotal evidence has been gathered during consultancy projects over a period of almost two years and involving a

diversity of clients in the manufacturing industry. The recurring theme that emerges is that detailed 3-dimensional animation of simulation results adds to the experience in several ways. In particular it provides:

- increased understanding of the system with less effort and technical knowledge required,
- the ability to include factory floor expertise in the process,
- added ability to appreciate spatial considerations,
- significant boost in management confidence in both the simulation and the proposed system,
- an excellent marketing tool for both internal and external use,
- a useful training tool for operations management and factory floor personnel,
- A generally useful aid to education about factory operations at all levels.

REFERENCES

- Bell, PC, Anderson, CK, Staples, DS and Elder, M (1999) 'Decision Makers Perceptions of the Value and Impact of Visual Interactive Modelling' *Omega, International Journal of Management Science* Vol 27 pp 155-65
- Kelton, WD, Sadowski, RP and Sadowski, DA (1998) *Simulation with Arena* Boston: McGraw-Hill
- Kirkpatrick PF and Bell PC (1989) 'Visual Interactive Modelling in Industry: Results from a Survey of Visual Interactive Model Builders' *Interfaces* Vol 19 No 5 pp 71-9
- Robinson, S (1994) *Successful Simulation* London: McGraw-Hill
- Van Dijk, JN, Jobing, MJ, Warren, JR, Seeley, D and Macri, R (1996) 'Visual Interactive Modelling with SimView for Organisational Improvement' *SIMULATION* Vol 67 No 2 pp106-20

SYSTEM DYNAMICS CONTINUOUS COMPUTER SIMULATION MODEL OF SHIPBUILDING PROCESS

Ante Munitic
Josko Dvornik
Split College of Maritime Studies
Zrinsko-Frankopanska 38, 21000 Split, Croatia
e-mail: munitic@pfst.hr, josko@pfst.hr

Slavko Simundic
Faculty of Law Split
Domovinskog rata 8, 21000 Split, Croatia
e-mail: slavko.simundic@pravst.hr

KEYWORDS

System dynamic modeling, continuous computer simulation, shipbuilding sub process, and business-production shipbuilding process.

ABSTRACT:

From 1990 to 1995 the authors have explored the existing organization structure of organization production construction model for ship «AMORELLA», especially sub process of «production preparing department». Result of their research was definition of the existing organization production model by using relatively new scientifically methodological technology of System Dynamics as a way of checking the validity of the model. System Dynamic Continuous Computer Simulation Model Of The Organization Production Shipbuilding Process– “Amorella”-PPBP, had a multilevel function of it's goal, from which the most important one's to mention are enabling business-production managing structure to continuously:

1. check the dynamics of shipbuilding process (costs and profit of the continuous shipbuilding process in whole, account view, payments of debits and credits, status of short terms and long terms bank credits, checking on the completing of working assignments of all of the executes of the shipbuilding process (cooperate firms and all working units of the shipyard),
2. efficiently gives the forecast of the possible effect of “bottleneck” in shipbuilding process;
3. optimizes the particular process parameters; and
4. finding the most valuable scenario of the future realization of the shipbuilding process.

1. CHARACTERISTIC OF BUISNESS – PRODUCTION SHIPBUILDING PROCESS

System Dynamic Computer Simulation Model of The Shipbuilding Process is a continuous model, which comprise: 1. qualitative (mental, verbal and structural model), and 2. quantitative (mathematical and computer model, behaviour dynamics model) model presented using DYNAMO SYMOBOLICS (diagram of material and informational flows) and as well in DYNAMO higher programming language, which is very compatible for

continuous use of business-production managing structure because of the:

1. Tracking and controlling of behaviour dynamics of PPBP (Abbreviation for term: Business-Production Shipbuilding Process (in Croatian: Poslovno proizvodnog Brodograđevnog Proces-PPBP)),
2. Forecasting the future behaviour of PPBP and
3. Optimization of parameters of PPBP.

The authors are using three main principles, which are: approximation and aggregation; System thinking philosophy, in order to present the Organization Business-Production Shipbuilding Process (PPBP) in following dynamic phases (flow of SP process) and in discrete control events (discrete event DD):

1. SP: SUPPLY OF SHIPBUILDING CAPACITY – SSC (Abbreviation for Term Supply Of Shipbuilding Capacity (in Croatian Ponuda brodograđevnog kapaciteta –PBK)) is now a part of global world marine market which requires: fresh market information on supply/demand fluctuation, obligate supply documentation, as well as information on market competition and information on client's financial standing (debits or credits).
2. D.D.: EFFICIENT COMPETITION TERMINATION, i.e. signing the agreement with consignee – SA (Abbreviation. Signing the Agreement), what initiate the sub process of PREPARING PROJECT DOCUMENTATION AND PREPARATION FOR OTHER PHASES. Also, material specification and production materials are needed, concluding contract with clients and cooperates, finishing deadlines and concluding deliveries, payments for debits and credits, and determination dynamics of sub-contractor employment.
3. Concluding the preparation of project documentation, which includes finishing the complete technological project documentation, begins the process of preparing sections and part of equipment out of slipway which requires the adequate documentation for reception warehouse that's completing is the basis for DD: setting (PUTTING DOWN THE KEEL -PK.)
4. Putting down the keel – PK is the basis for beginning of the shipbuilding on the slipway - PGBNN, beginning of the construction of the hull of the ship and continuously beginning of the other phases of ship constructor. In this phase it is very important to ensure the adequate documentation for worker's warehouses.

5. Finishing the shipbuilding on the slipway begins the process of DD: launching of the ship - PB, and then begins the process of SP: FINAL EQUIPPING OF THE SHIP - ZOB which requires the adequate equipment, and preparation for the controlling and documentation for transfer.

6. After phase of the equipping of the ship - ZOB follow DD: TRANSFER OF THE SHIP – PPB, which required collaudation documentation in case of possible finishing due to complaint of ship owner.

7. After transfer of the ship – PPB start SP: COMERCIAL USE OF THE SHIP IN THE GUARANTEE PERIOD – KKBUGR, which end in DD: when guarantee period run out – IGR.

8. At the end of guarantee period – IGR start SP: COMERCIAL USE OF THE SHIP – KKB. This phase comprise agreed time period (from 8 to 10 years), in which ship owner is obliged to fulfil all financial obligation (sum due) i.e. full cost price of ship toward shipyard. «Merchandise credit» shipyard => ship owner determine agreed time period.

9. DD: Speed (rate) in which shipyard collection debts - BNPB influence on decrease of SP: state of unpaid debts of shipyards – SNPOTB, whit final sum i.e. state equal zero.

2. EXAMPLE OF SYSTEM DYNAMICS MODEL OF 5th PHASE PPBP-a PB-LAUNCHING OF THE SHIP

It is evident that PPBP has characteristic of very complex management – intelligent system, and therefore software – module procedure is used for its realization. Global Simulation model of PPBP is presented with more than 250 analogical sub models, i.e. software modules with base structural model for every phase in PPBP, mutually analogues, shown in Figure 1.

a) Structural diagram of flows of 5th phase of PPBP

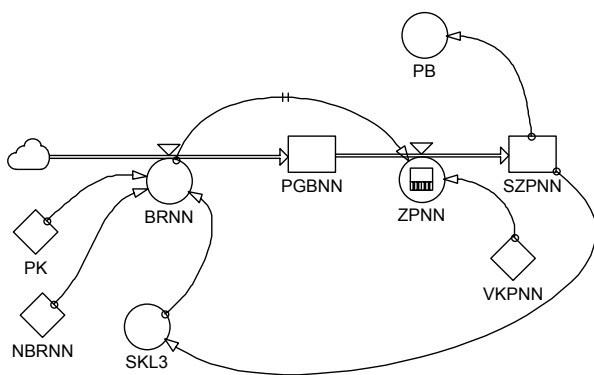


Figure 1. Structural diagram of flows of 5th phase of PPBP

b) Mathematical – computer model of 5th phase of PPBP

A PK.K=1
R BRNN.KL=PK.K*NBRNN*SKL3.K
C NBRNN=31712
L PGBNN.K=PGBNN.J+DT*(BRNN.JK- ZPNN.JK)

N PGBNN=0
R ZPNN.KL=DELAY3(BRNN.KL,VKPNN)
C VKPNN=30
L SZPNN.K=SZPNN.J+DT*(ZPNN.JK)
N SZPNN=0
A PB.K=CLIP(1,0,S302100.K+S302200.K+S102300.K+S302400.K+S202500.K+S103200.K+S109200.K+S109400+S1BSO.K+S1IRUS.K,SZPNN.K)*CLIP(1,0,S3S"A".K,PS3"A")*CLIP(1,0,S1S"B".K,PS1"B")*CLIP(1,0,SZPGBNN.K,NSZPGBNN)
C PS3"A"=24421000
C PS3"B"=3291000
A SKL3.K=CLIP(0,1,SZPNN.K,PSPNN)

Where are:

PK= placing down the keel (b.d.),

BRNN= speed of growth on the slipway (\$/day),

NBRNN= planed (standardize) labour on the slipway (\$/day),

PGBNN= beginning of building of the ship on the slipway (\$),

ZPNN= speed of finishing building of the ship on the slipway (\$/day),

VKPNN= time of completing production on the slipway (day),

SZPNN= cumulative state of finished labour on the slipway (\$),

PB= launching of the ship (b.d.),

PS3"A"= planed state of the warehouse "A" (\$),

PS3"B"= planed state of the warehouse "B" (\$),

SKL3= logical (expert) switch which control fulfilment of planed shipbuilding process, and from time to time stop it.

All others phases in shipbuilding process as active participation of all participant in shipbuilding production – business process are presented with analogue software module. This module function as “organisation whole”, with superadded sectors and sub sectors: outstanding debts, debts, transfer account, penalty, stimulations, total income, expenses, cost price, profit, basic funds, short terms and long terms loans. PPBP – model is programmed in DYNAMO and SYSDYNS higher programming languages.

For realization of the phase of launching the ship PB, all preliminary phases must be conditionally fulfilled in integrity: PBK- offer of shipyard capacities, which end with successfully signed contract for building the ship – PU, which previously state was PU=0, and with signed contract it changes state in PU=1. This initiates starting the phase of production documentation for realization and other equipment IIDIOP, and also specifications of materials and production materials, with signing the contract with suppliers and co-operations. Subsequently, starts making the sections and parts of equipments, supplying of entrance warehouse with corresponding documentation, which completeness is necessary for next phase PBPP i.e. putting down the keel – PK. If the phase IIDIOP is completed than it is possible to putt down the keel, and discrete variable PK is 1 (PK=0 is changed in PK=1), this starts next phase PPBP

i.e. PGBNN beginning of the shipbuilding on the slipway. For fulfilling launching of the ship previous phase PGBNN has to be fulfilled completely. If this condition is satisfied variable PB change its value PB=0 in new PB=1 and initialize next phase of PPBP i.e. final equipping of the ship ZOB.

3. CONCLUSION

System dynamic simulating modelling is one of the most appropriate and successful scientific dynamics modelling methods of the complex, non-linear, natural, technical and organizational systems.

Implementation of System dynamic continuous computer simulation sub model of the business-production shipbuilding process – PPBP, which is a part of discrete digital computer simulated models of continuous nonlinear realities, in BrodoSplit, has allowed it's managing structure the continuous application of «active simulation process» in archiving following effects:

1. Quantitative grading of historical equals of PPBP,
2. Narrowing the future uncertainty,
3. Completing the list of the costs of shipbuilding process of the actual process,
4. Heuristic optimization of PPBP and enlarging the financial stability,
5. Improvement of organization of the process and complete automation of PSBP.

REFERENCES

- Forrester, Jay W. 1973/1971. *Principles of Systems*, MIT Press, Cambridge Massachusetts, USA, 1973.
- Richardson, George P. and Pugh III Aleksander L. 1981, *Introduction to System Dynamics Modelling with Dynamo* MIT Press, Cambridge, Massachusetts, USA,
- Munitic, A. 1989. *Computer Simulation with Help of System Dynamics*, BIS, Croatia, 1989.
- Ante Munitic, Computer simulation of shipbuilding production process management,, *INTERNATIONAL CONFERENCE OF THE SYSTEM DYNAMICS SOCIETY*, JULY, 11-14-,1989., Germany,
- Ante Munitic, Slavko Simundic, System dynamics continuous computer simulation model of ship construction documentation elaboration subprocess, *IMAM'95*, 23-27 April, 1995., Croatia,
- Slavko Simundic, Ante Munitic, The knowledge basis model in the expert system structure for the technological preparation phase in shipbuilding, *IMAM'95*, 23-27 April, 1995., Croatia
- Ante Munitic, Slavko Simundic, (in Croatian), *Sistemski dinamički kontinuirani simulacijski model V faze brodogradevnog procesa "Porinuce broda"*, *SORTA*, 2002, Croatia.

BIOGRAPHY

Ante Munitic was born 08. 26.1941. in Omis, near Split, Croatia! He received his first BSc. in Electrotechnics Engineering in 1968, and his second BSc. in Electronics

Engineering in 1974; his MSc. degree in Electronics/Organization /Operational Research/Cybernetics Science in 1978, and his Ph.D. of Organization/Informatics Science (exactly: System Dynamics Simulation Modelling) in 1983. He is currently a University Professor of Information/Computer Science at the University of Split, Croatia. Prof. Munitic has published over 100 scientific papers on System Dynamics simulation modelling, operational research, marine automatic control system and The Theory of Chaos. He has published several books (as there are: "Computer Simulation with help of System Dynamics" and "Marine Electrotechnics and Electronics Engineering",). Today, he is professionally active university professor and scientist in the System Dynamics, Relativity Dynamics, System Dynamics Analogous Processes, Theory of Chaos and Informatics Scientific area.

Slavko Simundic was born in 1946. in Lovrec, Croatia. He is currently a University Professor of Informatics, Statistics and Basic methodology of social research using statistics at the Faculty of Law, University of Split, Croatia. Ph.D Slavko Simundic also teach the same course of lecture at The Police Academy – High Police School Zagreb, and on postgraduate study at Police Academy he teach Computer crime, Statistics and Functional Informatics. During his very productive career he was leader and coordinator of several scientific projects from file of Information and Organization Science in Economy. Since 1991. year he is leader of scientific project named “Modelling of organisation of preliminary production in maritime shipbuilding” financed by Croatia Ministry of science and technology.

Ph.D. Slavko Simundic is member of several international and domestic scientific and expert societies. Ph.D. Simundic has published over 250 scientific and expert papers from field of information and organization sciences, and he is also author of five monographs.

Josko Dvornik was born 1978. at Split Croatia, were he finished elementary and high Maritime school. In school year 1996/97 he enrolled Maritime University in Split, Marine engineering Department, completed all theoretical and practical subjects included in school program on time, and passed all exams. He graduated in 2000. year on theme “*Application on computer simulation dynamics of behavior of ship propulsion system: windlass – asynchronous engine*”, with very good degree as a first student in his class. Since December 2001. year he has worked as younger assistant at Maritime University in Split on scientific project titled “Computer simulation model of maritime educative system of Croatia”. In June 2002. he has enrolled postgraduate study of engineering at Faculty of Mechanical Engineering and Naval Architecture. He has published 20 scientific papers on System Dynamics simulation modeling.

CHEMICAL ENGINEERING MANAGEMENT

SIMULATION OF RUBBER MOULDING USING FINITE ELEMENT MODELLING

Vahid Nassehi *

Babar Ali

Chemical Engineering Department

Loughborough University

Loughborough LE11 3TU, UK

v.nassehi@lboro.ac.uk

KEYWORDS

Modelling, injection moulding, free surface flow

ABSTRACT

Finite element technique is used to model the free surface viscoelastic flow regime representing injection mould filling of elastomeric material such as rubber compounds. The most distinct feature of the constitutive behaviour of elastomers, is the influence of material elasticity on the elongation and shear deformation suffered by the fluid during flow. In the present study this feature has been tackled through the use of Phan-Thien/Tanner model. A scheme based on the Volume of Fluid technique is developed for free surface tracking. In conjunction with this scheme a de-coupled algorithm, in which the flow and constitutive equations are solved in separate steps, is utilised to reduce the computational cost of simulations. The results represented in this paper are in agreement with the theoretical expectations and prove the applicability of the developed model to industrially relevant situations.

INTRODUCTION

Computer modelling has been extensively used in the simulation of injection moulding of thermoplastics (Isayev, 1991). However, despite the wide spread application of modelling in polymer processing simulation of rubber moulding has received relatively little attention. This is because that modelling strategies developed for purely viscous fluids do not yield accurate results for processes involving elastomers. These processes are often significantly influenced by phenomena such as elongational deformation of the fluid, as well as shearing effects. Representation of such phenomena in mathematical flow models is not straightforward. For example, there is an extensive body of published data indicating that commonly used constitutive equations such as the Maxwell class of viscoelastic models do not provide acceptable simulations for processes such as injection moulding which are influenced by elongational behaviour of rubber compounds (Tanner, 2000). Therefore complex viscoelastic behaviour of rubber is often given as the main source of difficulty in generating useful simulations for rubber moulding. There is no doubt that simulation of rubber moulding should be based on

rheologically sound equations. However, there is another phenomenon besides the rheological factors which can potentially determine the outcome of rubber moulding process. This factor is the possibility of the onset of premature crosslinking of rubber molecules at the free surface of the fluid flow front entering the mould. Such crosslinking will alter the material properties of the fluid and may harden the flow front causing an impediment for the flow. The premature or short shot crosslinking can therefore interfere with mould filling and result in the appearance of cavities and gaps within the moulded compound. In this paper we present a finite element model for the simulation of injection moulding of rubber. This model is based on a rheological model which is shown to represent both the elongational and shear behaviour of elastomers. The advancement of the free surface into a mould cavity is handled via a robust moving boundary tracking method. It is shown that the developed model is capable of generating detailed results for mould filling of rubber and can be used to evaluate the influence of various factors on the outcome of the process. Thus the constructed model, in conjunction with appropriate experimental measurements, provides a cost effective tool for designing injection moulding operations of rubber compounds. Brief discussion of the governing equations of the present model and an outline of the developed solution strategy are given in the next section.

GOVERNING MODEL EQUATIONS USED IN THE PRESENT WORK

Using a two-dimensional coordinate system (x, y) the main model equations consist of the mass continuity for incompressible fluids given as

$$\frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} = 0 \quad (1)$$

and the equation of motion for non-Newtonian fluids which in the absence of body forces is written as (Cauchy's equation of motion)

$$\begin{cases} \rho \left(\frac{\partial V_x}{\partial t} + V_x \frac{\partial V_x}{\partial x} + V_y \frac{\partial V_x}{\partial y} \right) = -\frac{\partial P}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} \\ \rho \left(\frac{\partial V_y}{\partial t} + V_x \frac{\partial V_y}{\partial x} + V_y \frac{\partial V_y}{\partial y} \right) = -\frac{\partial P}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} \end{cases} \quad (2)$$

where V_x and V_y are the components of the velocity field, ρ is the fluid density, P is the pressure and τ_{xx} etc. are the components of the extra stress tensor. For purely viscous fluids (i.e. generalised Newtonian fluids) which do not exhibit significant elongational behaviour, it is possible to relate explicitly the components of the stress appearing in the equation of motion to the rate of strain (or deformation) within the fluid. This results in the elimination of the stress components from the equation of motion. However, for highly elastic fluids such as rubber this is not possible and in order to take into account the true constitutive behaviour of such fluids implicit relationships between the stress and the rate of strain components should be used. Therefore in this work the stresses appearing in equation (2) are related to the rate of strain tensor using the following constitutive equation, developed by Phan-Thien and Tanner (1997), which can realistically represent the behaviour of viscoelastic fluids.

$$\left\{ \begin{aligned} & \lambda \left(\frac{\partial \tau_{xx}}{\partial t} + V_x \frac{\partial \tau_{xx}}{\partial x} + V_y \frac{\partial \tau_{xx}}{\partial y} \right) - 2\lambda \left(\left(1 - \xi \right) \frac{\partial V_x}{\partial x} \tau_{xx} + \left(\frac{\partial V_x}{\partial y} - \frac{\xi}{2} \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right) \right) \tau_{xy} \right) \\ & + \left(\left(1 + \frac{\varepsilon \lambda}{\eta^o} (\tau_{xx} + \tau_{yy}) \right) \right) \tau_{xx} = 2\eta \frac{\partial V_x}{\partial x} \\ & \lambda \left(\frac{\partial \tau_{xy}}{\partial t} + V_x \frac{\partial \tau_{xy}}{\partial x} + V_y \frac{\partial \tau_{xy}}{\partial y} \right) - \lambda \left(\frac{\partial V_x}{\partial y} - \frac{\xi}{2} \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right) \right) \tau_{yy} \\ & - \lambda \left(\frac{\partial V_y}{\partial x} - \frac{\xi}{2} \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right) \right) \tau_{xx} \\ & + \left(\left(1 + \frac{\varepsilon \lambda}{\eta^o} (\tau_{xx} + \tau_{yy}) \right) \right) \tau_{xy} = \eta \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right) \\ & \lambda \left(\frac{\partial \tau_{yy}}{\partial t} + V_x \frac{\partial \tau_{yy}}{\partial x} + V_y \frac{\partial \tau_{yy}}{\partial y} \right) - 2\lambda \left(\left(1 - \xi \right) \frac{\partial V_y}{\partial y} \tau_{yy} + \left(\frac{\partial V_y}{\partial x} - \frac{\xi}{2} \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right) \right) \tau_{xy} \right) \\ & + \left(\left(1 + \frac{\varepsilon \lambda}{\eta^o} (\tau_{xx} + \tau_{yy}) \right) \right) \tau_{yy} = 2\eta \frac{\partial V_y}{\partial y} \end{aligned} \right.$$

where ε is a characteristic elongational parameter, ξ , ($0 \leq \xi \leq 2$) is representative of internal shearing behaviour of the fluid, λ is relaxation time, η is shear viscosity given by the power-law model as $\eta = \eta^o \left(\dot{\gamma} \right)^{n-1}$,

where, η^o is the material consistency coefficient, n is the power-law index and $\dot{\gamma}$ is the shear rate given by

$$\dot{\gamma} = \left\{ 2 \left(\frac{\partial V_x}{\partial x} \right)^2 + 2 \left[\left(\frac{\partial V_y}{\partial y} \right)^2 + \left(\frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \right)^2 \right] \right\}^{\frac{1}{2}}$$

MODELLING STRATEGY

We have developed a de-coupled solution strategy in order to solve the above described model equations. In this approach the simulation cycle starts with the solution of Stokes (i.e. Newtonian) equations for incompressible fluids. This solution yields the first set of velocity and pressure fields that can be inserted into the P-T/T equation to obtain the viscoelastic stress components. Stress gradients are in turn found and inserted into the Cauchy's equation and new velocity and pressure fields are computed. The cycle consisting of the solution of the P-T/T and Cauchy's is iterated until convergence is achieved. Time variable is then incremented and the cycle is repeated till the predetermined end of simulation. Solution of the Stokes flow equations at the start of the simulation is based on the continuous penalty/ Galerkin finite element scheme which combines efficiency with computing economy (Nassehi, 2002). A similar scheme is used to solve the Cauchy's equation of motion. The velocity field obtained using the continuous penalty scheme provides the necessary data for generating the pressure field via the variational recovery method (Pittman and Nakazawa, 1984). The variational recovery method is also used to calculate gradients of all field variables (e.g. stress gradients) at computational nodes. The solution of the free surface equation, which mainly represents the convection of free surface boundary in line with the flow, is based on the Stream Line Upwind Petrov Galerkin (SUPG) scheme (Nassehi, 2002). This equation is solved at the end of each time step after converged values for the velocity are found to determine the position of the free front. Time stepping used in conjunction with all of these schemes is based on the well known implicit θ method (Nassehi, 2002). Details of the derivation of the working equations of the described scheme are published previously (Hou and Nassehi, 2001) and will not be repeated here. The main focus of the present study has been the extension of this scheme to free surface flows which is the dominant factor in the simulation of mould filling processes.

FREE SURFACE TRACKING

The advancing flow front within the mould is simulated using the VOF (volume of fluid) technique. This method is based on the solution of the surface position probability density equation, given as

$$\frac{\partial F}{\partial t} + V_x \frac{\partial F}{\partial x} + V_y \frac{\partial F}{\partial y} = 0 \quad (3)$$

where $0 \leq F \leq 1$ is called the surface position function. In this work we have used a modified version of the VOF technique in which the moving boundary flow regime is considered as a two-phase regime in which the filled and the voids (air filled) sections are considered as different phases (Thompson, 1986). The flow model is solved for the entire domain whilst at each section physical properties relevant to that phase are inserted. Simultaneous solution of equation (3) with the flow model generates values of F . A value of F between 0 and 1 (usually $F = 0.5$) is taken as the boundary between phases representing the moving free surface. Values of physical parameters for each phase in the flow

field is related to the position of the free surface using the following equation

$$y = y_f F + y_a (1 - F)$$

where y is a given physical parameter and y_f and y_a are the values of this parameter in the fluid and air filled regions, respectively, (Nassehi and Ghoreishy, 1997).

FINITE ELEMENT DISCRETISATION OF THE FREE SURFACE EQUATION AND SOLUTION ALGORITHM

We have used the weighted residual finite element method to derive the working equation corresponding to the free surface position function. In this procedure the inner product of equation (3) and a suitable weight function is constructed as

$$\langle \dot{F}, \hat{\psi}_i \rangle + \bar{V} \bullet \nabla F, \hat{\psi}_i \rangle = 0 \quad (4)$$

where $\hat{\psi}_i$ represents streamlined upwind weight function given as

$$\hat{\psi}_i = \varphi_i + \gamma_c \frac{|h_x V_x + h_y V_y|}{2|\bar{V}|^2} \left(V_x \frac{\partial \varphi_i}{\partial x} + V_y \frac{\partial \varphi_i}{\partial y} \right)$$

where φ_i is the normal Galerkin weight function which is identical to the shape functions used to represent elemental

approximation of the function F as $\tilde{F} = \sum_{j=1}^p F(j) \varphi_j$, p is

the number of nodes per element, $0 < \gamma_c \leq 1$ is the upwinding constant, and h_x and h_y are characteristic lengths given as

$$h = \sqrt{(h_x)^2 + (h_y)^2} = \sqrt{\left(\frac{\partial x}{\partial \xi} + \frac{\partial x}{\partial \eta} \right)^2 + \left(\frac{\partial y}{\partial \xi} + \frac{\partial y}{\partial \eta} \right)^2}$$

where (ξ, η) is a elementally defined co-ordinate system. Within the space of a finite element after substitution of function F with approximate representation in terms of elemental shape functions equation (4) is written as

$$\int_e \left[\sum_{j=1}^p \dot{F}_j \varphi_j + \left(V_x \frac{\partial}{\partial x} F_j \varphi_j + V_y \frac{\partial}{\partial y} F_j \varphi_j \right) \right] \hat{\psi}_i dx dy = 0 \quad i = 1, \dots, p$$

Using matrix notation

$$[M] \{\dot{F}\} + [K] \{F\} = 0 \quad (5)$$

where $[M]$ and $[K]$ are the mass and stiffness matrices, respectively. After the time stepping of equation (5), via the θ method (Nassehi, 2002), the required working equation is obtained as

$$([M] + \theta \Delta [K]) \{F\}_{n+1} = ([M] - (1 - \theta) \Delta [K]) \{F\}_n$$

where Δt and n represent time increment and time step, respectively.

The algorithm used for the free surface tracking can be summarised as

1. The entire domain of interest is first discretised into a mesh of finite elements.
2. An initial configuration representing the partially filled domain is considered and an array consisting of the appropriate values of $F=1, 0.5, 0$, for the nodes containing fluid, free surface boundary, and air, respectively, is prepared. The sets of initial values for the nodal velocity and pressure fields in the solution domain are assumed and stored as input arrays. An array containing the boundary conditions along the external boundaries of the solution domain is prepared and stored.
3. The time variable is updated, incrementing it by Δt .
4. It is initially assumed that the flow field in the entire domain is incompressible and using the initial and boundary conditions, the corresponding flow and rheological equations are solved to obtain the velocity, pressure and stress distributions.
5. New values of the material parameters such as the shear viscosity are determined using the computed velocity field.
6. Steps 4-6 are iterated until the solution cycle converges. Air filled regions having a calculated pressure more than a given threshold are identified. Inside these regions volume of the air is recalculated using equation of state for ideal gases. This allows the adjustment of the volume of air bubbles trapped inside rubber matrix by relating it to the calculated pressure. Air filled sections that are adjacent to the outer boundaries of the mould are not affected by this adjustment because in these regions air can freely flow out of the domain without being compressed. Therefore pressure in these sections do not rise above the given threshold and volume adjustment in these regions is not necessary.
7. The converged velocity field is used to solve the free surface equation.
8. Using the updated values of the free surface function, the location of the free surfaces are identified and the position of each phase in the current flow domain is marked accordingly.
9. If the current time has reached the final time required for the flow simulation then the calculations are terminated, otherwise steps 3-9 in the above algorithm are repeated and the solution is advanced in time.

COMPUTER SIMULATIONS

The mould used to study these effects is of two-plate design whose schematic diagram is shown in figure 1. The feed system for this mould consists of a sprue connected to a U shaped runner which ends at a diverging gate. The material is injected through sprue at a constant flow rate using an injector. The injector is a displacement device which is designed to maintain a steady flow rate entering into the runner through the sprue. The injection continues until the mould is filled.

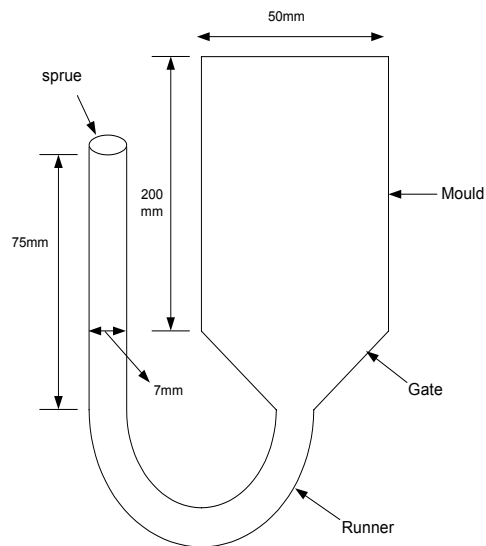


Fig 1 Schematic Diagram of Injection Moulding Process Simulated Here

The input data used in the present simulation are:

Physical data

rubber density = 980 kg m^{-3} , consistency coefficient = $88700 \text{ kg m}^{-1} \text{ s}^{-1}$, power law index = 0.28, relaxation time = 1.5 s, elongation parameter (ε) is taken to be 10 times higher than the shear viscosity, Clarke and Petera (1997), shear dependency parameter in the P-T/T model $\xi = 1$, density of air = 1.2929 kg m^{-3} , viscosity of air = $0.251 \times 10^{-4} \text{ Pa s}$.

Initial and Boundary conditions

Initial velocity=0, everywhere. Inlet velocity= 0.045 m s^{-1} (corresponding to a constant volumetric flow rate). No slip at solid walls. At the exit stress free conditions are imposed (velocity gradients and stresses are set to be zero).

Data used in the finite element scheme

Penalty parameter $\lambda = 10^{10}$, Time increment $\Delta t = 0.0245 \text{ s}$.

Initially, the flow domain is discretised into 385 nine noded C^0 bi-quadratic finite elements and the convergence of the solutions is checked via mesh refinement by doubling the number of elements after the initial run. At the end of the first time step stresses at the solid walls are computed and old time step values are inserted as the boundary conditions for the P-T/T equation to eliminate the contradiction between the set of conditions used for the flow and constitutive components of the model.

The mould filling operation considered here takes 2.5 s in total. Choosing a time increment of 0.0245 s the process is simulated through nearly 100 steps. In the following figures simulated flow front within the modelled mould cavity obtained at various time steps are shown. Accuracy of the simulations is checked by comparing amount of fluid entered into the mould with the fluid injected through the sprue at a steady rate. The amount of fluid entering the mould can be found by measuring the extent of the advancement of the flow front.

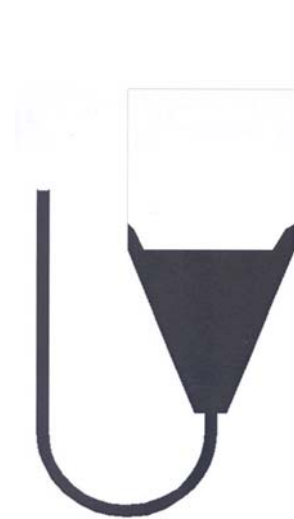


Fig 2 Predicted Free Surface Position After the 1st Time Step

As the following sample results show the advancement of the free surface boundary is tracked using the described algorithm



Fig 3 Predicted Free Surface Position After the 25th Time Step



Fig 4 Predicted Free Surface Position After the 51st Time Step



Fig 5 Predicted Free Surface Position After the 90th Time Step

Finally at the end of the injection process the filled cavity which contains an air bubble is simulated. This shows the inefficiency of the design and operational conditions used in the example which is the subject of present simulations.



Fig 6 Predicted Free Surface Position After the 100th Time Step

CONCLUSIONS

The developed finite element scheme is shown to generate useful simulations for the mould filling process. Despite using a dominantly hyperbolic governing rheological equation, which is employed to take into account both the elongational and shear behaviour of elastomeric fluid being modelled, stable numerical results are generated without excessive computational cost. The observed asymmetry of the flow front in the results can be attributed to various factors such as highly non-linear nature of the viscoelastic fluid. Such a fluid is expected to demonstrate an uneven deformation on its parts during the flow through the U shaped runner modelled in the present study. Overall, the results are self consistent and provide useful data for further analysis of the mould filling process. The work is now being extended to examine the effect of crosslinking of rubber chains on the free surface by comparing the above results

with simulations obtained using material data for crosslinked rubber within elements located at the flow front.

ACKNOWLEDGMENT

The authors wish to thank Engineering and Physical Research Council of UK (EPSRC) for a grant which enabled the development of the finite element model used in this work.

REFERENCES

- Clarke, J. and J. Petera. 1997. "Determination of steady state elongational viscosity for rubber compounds using bell mouth dies." *J. Applied Polymer Sciences*, Vol. 66, 1139.
- Hou, L. and Nassehi, V. 2001 "Evaluation of stress-effective flow in rubber mixing", *Nonlinear Analysis*, Vol. 47, 1809.
- Isayev, A.I.; 1991. *Modelling of Polymer Processing*. Hanser Publisher, Munich.
- Nassehi, V. and M.H.R. Ghoreishy. 1997. "Simulation of free surface flow in partially filled internal mixers." *International Polymer Processing*, Vol.12, 346.
- Nassehi, V.; 2002. *Practical Aspects of Finite Element Modelling of Polymer Processing*. John Wiley and Sons, Chichester.
- Pittman, J.F.T.; and S. Nakazawa. 1984. *Finite Element Analysis of Polymer Processing Operations*. In: Pittman, J.F.T., Zienkiewicz, O.C., Wood, R.D. and Alexander, J.M., Numerical Analysis of Forming Process. John Wiley and Sons, Chichester.
- Phan-Thien, N. and R.I. Tanner. 1977, "A New Constitutive Equation Derived From Network Theory." *J. Non-Newtonian Fluid Mech.* Vol.2, 353.
- Tanner, R.I.; 2000. *Engineering Rheology*, 2nd Ed. Oxford University Press, Oxford.
- Thompson, E. 1986. "Use of Pseudo-concentration to follow creeping flows during transient analysis." *Int. J. Numer. Methods Fluids*. Vol.6, 749.

BIOGRAPHY

Vahid Nassehi obtained his first MSc, in chemical engineering, at the university of Teheran, Iran, in 1974. He worked as a research and development engineer in the oil industry in Iran until 1976 and after that continued his studies at the University of Wales, UK where he was awarded degrees of MSc (1978) and PhD (1981). Before joining Loughborough University in 1989 he worked as a post-doctoral research assistant in the University college of Swansea and later as senior lecturer in the Teesside Polytechnic. His main area of interest is computer aided analysis and design of engineering flow systems. Currently, he holds the chair of computational modelling at the chemical engineering department of Loughborough University, UK.

*Corresponding Author:

Professor Vahid Nassehi, Chemical Engineering Department, Loughborough University, Loughborough LE11 3TU, UK
v.nassehi@lboro.ac.uk
 Tel +44 (0) 1509 222522
 Fax +44 (0) 1509 223923

OPTIMIZATION OF TRANSFER AND STORAGE OPERATIONS IN REFINERIES

Antonio Eduardo Matsuno Ramos¹

Lúcia Valéria Ramos de Arruda²

Flávio Neves Jr.³

Laboratório de Automação de Sistemas de Controle Avançado, CPGEI

Centro Federal de Educação Tecnológica do Paraná

Av. Sete de Setembro, 3165

80230-901 Curitiba-PR-Brazil

E-mails: ramos@cpgei.cefetpr.br¹; arruda@cpgei.cefetpr.br²; neves@cpgei.cefetpr.br³

KEYWORDS

Decision Support Systems, Optimization, Path Selection, Transfer and Storage, Refinery

ABSTRACT

This paper presents a path selection methodology, which helps a refinery operator to choose routes in transfer and storage operations. These operations are common tasks in refineries and their optimization will certainly increase the refinery profitability. All refineries have a tank farm, where products (like crude oil, gasoil, diesel, kerosene, etc.) are stored. The tanks are interconnected through a pipe network, which allows the transferring of products among the storage tanks. To make a transfer operation possible, a route between source and destiny tanks must be selected and, due to the network complexity, the optimal path is not easily found. The methodology presented in this paper can obtain valid transfer routes, showing them in a computer-graphical-interface. Based on an optimization criterion, the best transfer route can also be obtained, and it is highlighted in a graphical interface. This interface can dramatically help the decision-making process of refinery operators.

INTRODUCTION

Nowadays, the world economy is highly affected by the petroleum industry. The development of global competition has made the chemical processing industry search ways of increasing productivity and reducing operational costs in order to compete in this scenario.

Optimization techniques for refinery operations can highly increase the oil industry profitability (Pinto et. al., 2000). This paper discusses the optimization of transfer and storage operations, which are common tasks in refineries. Daily, in a refinery, it is necessary to move products from pipelines to tanks, and vice versa. Sometimes, restorage is demanded, and tank to tank movements are executed.

In a refinery, all the product storage and movement is made in the tank farm, which is responsible for all the supply and discharge activities, as well as blending and storage of finished products (Kuttelwascher and Rhemann, 1995).

The tank farm contains tanks (which store products), and valves and pumps (which control the product movements). In order to execute a transfer operation, it is necessary to choose a valid piping route between the source and the destiny locations. In typical refinery tank farms, the number of oil movement patterns can amount to several thousand (Takahashi and Kateeshock, 1992). Due to the pipe network complexity, refinery operators just seek a valid path, without considering any optimization criterion that could lead to a better path selection. The major problems in this task are contamination and pipe or valve damages. Contaminations occur when two incompatible products are mixed, making one of them (or both) lose its commercial value. Damages may occur when valves that should be opened are closed, and a pump forces product movement through them.

The path selection methodology presented in this paper considers the refinery tank farm as a graph and shows possible paths between source and destiny spots. Two optimization criteria are evaluated: the number of valve operations and the volume of pipes needed to execute a transfer operation.

OVERVIEW

A refinery tank farm contains several tanks, interconnected by pipes. Pumps move products, and valves define their trajectories. There is at least one pipeline that supplies crude oil from outside. Many refineries also use pipelines to discharge finished products. There is also a load facility, where finished products are discharged in trucks. Figure 1 shows an overview of a typical refinery tank farm.

A more accurate view of the connections among the tanks is shown in Fig. 2. Every tank has one (sometimes more than one) valve directly connected to it, which allows supply and discharge. This valve is linked to other valves, which are connected to the main pipes of the refinery network. An ensemble of pipe bifurcations is called **manifold**. There are four manifold connected to tanks in Fig. 2 (the first one is limited by V1/V11/V12/V13/V14, the second one is limited by V2/V21/V22/V23/V24, the third one is limited by V3/V31/V32/V33/V34, and the last one is limited by V4/V41/V42/V43/V44).

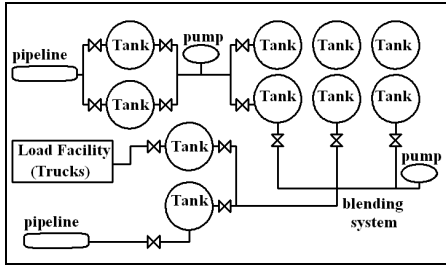


Figure 1: Overview of a Tank Farm

Every pump has at least two valves, one for supply, and other for discharge. Some pumps have more than one supply valve.

The numbers on the top of Fig. 2 show the coordinates of the tanks. Numerical values in Fig. 2 are illustrative and do not correspond to an actual tank farm dimension.

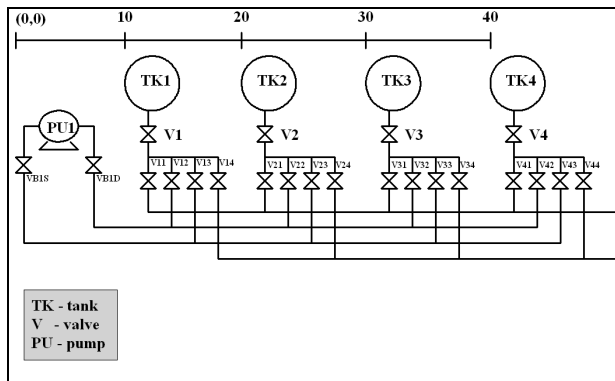


Figure 2: Accurate View of Tank Connections

During the path selection process, an operator must open some valves to allow a product flow through them and close others to avoid contaminations. In some cases, one or more pumps must be turned on to make products move through pipes. A pump is not needed in some restorage operations (tank to tank movement), when the product level of the source tank is higher than the product level of destiny tank. For example, when transferring from TK3 to TK4 (not using pumps), an operator could open V3, V32, V42, and V4, and close V31, V33, V34, V41, V43, V44, V22, V12, and VB1D. Although V12, V22, and VB1D are far from the source and destiny tanks, they can't be ignored, or a contamination occurs in TK1 and TK2.

MODELING

In this paper, an object oriented approach was used to model the tank farm network of a refinery. This technique provides tools for constructing generic, extensible and adaptable systems. McCormack and Roberts (1996) successfully applied object oriented programming in a similar problem, whose objective was finding routes in private road, bus, and

rail transport systems. Object oriented modeling is efficient to simulate large networks, like tank farms. Mikler et. al. (1998) applied this methodology to simulate large communication networks. Also, Wöhlhah and Engell (1995) used an object oriented approach to model and simulate chemical and biochemical plants.

Figure 3 shows six classes, which were defined to represent the tank farm. The classes Valve, Tank, Pump, Junction, and Pipe contain specific information related to the most important components in a tank farm. The class General contains information related to all components, like *id* (a string to identify each component), and *xcoordinate* and *ycoordinate* (integers that inform the position of the component in a tank farm).

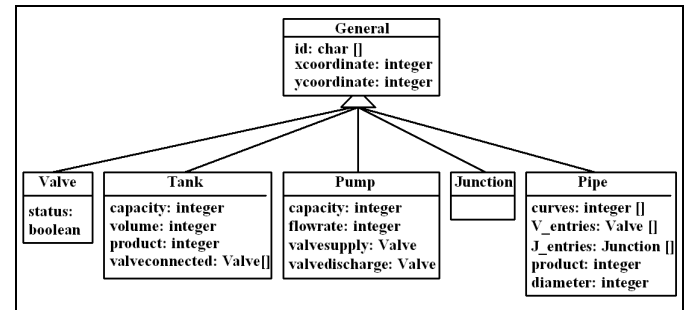


Figure 3: Classes

Once the classes are defined, it is possible to create objects. In order to reproduce the tank farm shown in Fig. 2, four objects from Tank class should be created. Their *id* should be TK1, TK2, TK3, and TK4; their *xcoordinate* should be 10, 20, 30, and 40.

Valve class

This class contains only one attribute, and all objects from this class must be created before the others. The user must inform the *id*, the coordinates, and the *status* to create Valve objects. *Id* is needed to identify a valve in the network. The coordinates are used to inform the operator the position of the valve. It is important to emphasize that, in many refineries, not all the valves can be electronically operated, what demands an operator go to the valve spot and operate it manually. No other value but OPEN or CLOSED is allowed in the *status* field.

Tank class

This class provides information about the attributes of refinery tanks. *Capacity* informs the maximum volume allowed inside the tank, *volume* informs the current volume stored in it, and *product* informs which product is stored. In the attribute *valveconnected*, Valve objects previously created must be informed.

Pump class

The most important attributes of this class are the valves directly connected to the pump, its capacity, and current flow rate. Capacity means the maximum flow rate that the pump can work at. In a normal operational condition, the pumps work either at the maximum flow rate or remain switched off. In the attributes related to the valves, the user must specify Valve objects previously created.

Junction class

This is the simplest class and the only attributes needed to describe a junction are the ones defined in the class General. Junctions are used to connect two Pipe objects.

Pipe class

This is the most complex class, and represents the pipes, which connect all components defined by the three first classes previously described in this session.

One of the difficulties is to define what a pipe is. There are many lines in Fig. 2, and a simple definition would consider a pipe as a line in the figure. However, the left side of Fig. 4 shows a part of Fig. 2 that would require five pipes to describe a manifold, as shown in the right side of Fig. 4. Also, it would require four Junction objects to represent the connections between two pipes. To reduce the number of objects, in this paper, **a pipe is an ensemble of lines, limited by valves and/or junctions**. Using this definition, there is only one pipe and no junctions in Fig. 4.

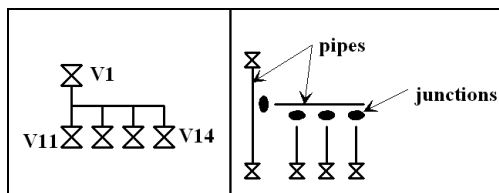


Figure 4: Pipe Representation

Pipes have a main line, in which the valves and junctions are connected. The main line can be straight or segmented, and is defined by coordinated pairs (x,y). The left side of Fig. 5 shows a pipe arrangement and five valves, and the right side shows how this arrangement can be represented using two Pipe objects (Pipe *a* and Pipe *b*), five Valve objects (VA to VE), and one Junction object (Junction *j*).

In the right side of Fig. 5, pipe main lines are shown in continuous drawings. Pipe *a* main line is straight, and two coordinated pairs are enough to represent it. Pipe *b* main line contains three segments, and requires four coordinated pairs to define it.

Valves VB, VC, and VD, and Junction *j* are connected to Pipe *a* main line. Valve VA is connected to the first segment

of Pipe *b* main line, and Valve VE and Junction *j* are connected to its third segment. Creating these objects and making their connections, the arrangement shown in Fig. 5 can be successfully represented.

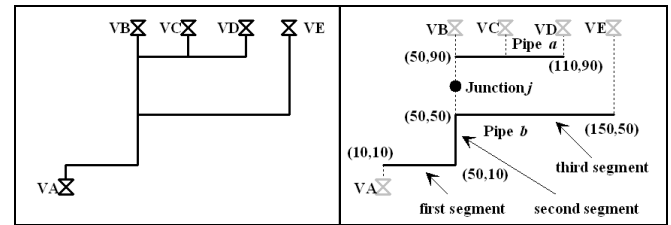


Figure 5: Pipe Arrangement Representation

Using the definition of pipe previously stated, Fig. 2 contains only eight pipes (P1 to P8), as shown in Fig. 6. The continuous drawings show the main lines of the pipes, and the dotted lines show the valve entries. No junctions were required to describe pipe arrangements in Fig. 2.

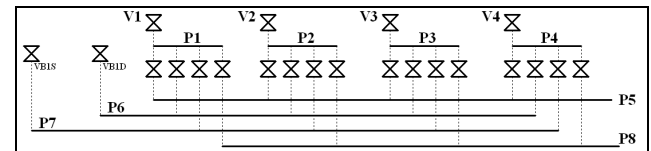


Figure 6: Pipes in the Tank Farm

The main attributes of a pipe are a list of coordinated pairs (at least two) that defines the main line, a list of valves (the user must specify Valve objects), a list of junctions (Junction objects must be specified), the diameter, and the product stored in the pipe. It is important to notice that pipes always operate completely filled, thus, there is always a product remain inside them.

To calculate the pipe volume, the main line is considered as being a cylinder, whose height is the main line length and diameter is informed by the user. Volumes between the main line and the valves must be added to the main line volume in order to obtain the total volume. In Fig. 7, the volume of the pipe P1 (also represented in Fig. 6) is the sum of the volumes of the six cylinders shown in the Fig. 7.

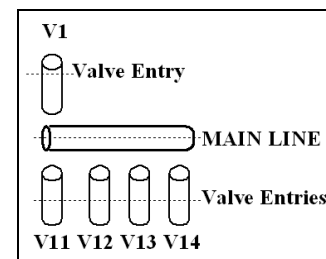


Figure 7: Volume of a Pipe

It is important to observe that when a transfer operation is made, the product in movement fills the whole pipe, and remains there after the operation is finished. When transferring a product, paths that contain pipes filled with incompatible products must be ignored.

PATH SELECTION

In order to select paths, the tank farm is converted to a graph $G(V,E)$. V is the set of vertices (or nodes), and E is the set of edges (or links). Tanks, pumps, and pipes are converted to vertices, and valves become the edges of G . Two or more pipes connected by junctions must be converted to a single node. Figure 8 represents the tank farm shown in Fig. 2, as a graph.

In a transfer operation, the source spot can be a pipeline or a tank, and the destiny spot can be a pipeline, a tank, or a load facility. In all cases, source and destiny are always vertices. Edges are connections between two vertices, and can be bi-directional or unidirectional. Valves connected to tanks are bi-directional, because they must allow both supply and discharge operations. Valves connected to pumps are unidirectional, because a pump only operates in one-way direction. In fact, the valves in refineries can be classified as bi-directional, except the ones that are directly connected to pumps (VB1D and VB1S, in Fig. 8).

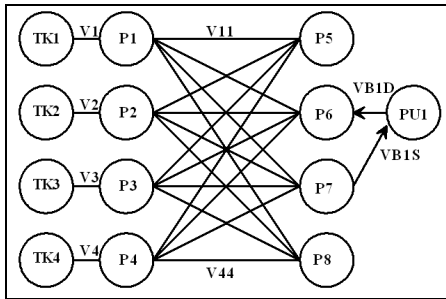


Figure 8: Tank Farm Shown as a Graph

Once the tank farm has become a graph, methods to find routes in graphs can be applied.

To optimize path selections, a cost must be defined to each vertex. To calculate vertex costs, an optimization criterion must be chosen. Two optimization criteria are proposed in this paper:

- number of valve operations;
- total pipe volume.

A valve operation is the action of opening a closed valve or closing an open valve. In a simulation scenario where many valves need to be manually operated, the number of valve operations becomes an important optimization criterion due to the workforce needed to develop such task.

Total pipe volume is the sum of volumes of all the pipes used in the operation. Pipe volumes are calculated as shown in Fig. 7.

Two algorithms were tested to select paths. The first one is a recursive algorithm, which looks for an edge connected to the source vertex and checks the other node connected to this edge. This new vertex is considered as source node, and search continues until destiny is reached, as shown in Fig. 9.

```
Search (Node source, Node destiny, integer i)
If source  $\neq$  destiny then
  For j = 1 to i
    If path[j] == source then return;
  path[j] = source;
  volumecost = volumecost+cost(source);
  For j = 1 to number_of_valves(source)
    valvecost2 = valvecost;
    For k = 1 to number_of_valves(source)
      If k  $\neq$  j and open(getvalve(source,k)) then
        valvecost = valvecost+1;
        insert_closetlist(getvalve(source,k));
        valve = getvalve(source,j);
        If closed(valve) then
          valvecost = valvecost+1;
          insert_openlist(valve);
        source = get_other_node(valve,source);
        Search(source,destiny,i+1);
        valvecost = valvecost2;
    volumecost = volumecost-cost(source);
  Else return path, volumecost, valvecost;

(parameter i must be zero at first function call)
```

Figure 9: Path Selection Algorithm

Table 1 describes variables and functions found in the algorithms.

Table 1: Variables and Functions

Variables and Functions	Description
path	Array containing nodes between source and destiny
valvecost	Number of valve operations needed to perform transfer
valvecost2	Auxiliary variable used to recover value of <i>valvecost</i> after a Search call is made
volumecost	Sum of volume of all pipes between source and destiny
Search(Node S, Node D, integer i)	Search paths from source node S to destiny node D
cost (Pipe P)	Returns the volume of P
number_of_valves(Node N)	Returns the number of valves connected to N
open(Valve V)	Returns TRUE when V is open, FALSE otherwise
closed(Valve V)	Returns TRUE when V is closed, FALSE otherwise
getvalve(Node N, integer j)	Returns the <i>j</i> -th valve connected to N
get_other_node(Valve V, Node N)	Returns a node connected to V, different from N
insert_openlist(Valve V)	Inserts V in the list of valves that must be opened
insert_closetlist(Valve V)	Inserts V in the list of valves that must be closed

The second algorithm is Dijkstra's (Dijkstra, 1959), an efficient algorithm for constrained path selection (Korkmaz

and Krunz, 2001). Solka et. al. (1995) and Boroujerdi and Uhlmann (1998) also described Dijkstra's algorithm efficiency in fast computation of optimal paths. However, Dijkstra's algorithm (described in Fig. 10) supports edge costs, not vertex costs, and requires an $N \times N$ adjacency matrix A , where N is the number of vertices in a graph. A_{ij} is the cost to go from node i to node j .

```

Variables used in Dijkstra's algorithm:
A - adjacency matrix
N - number of vertices
C1, C2, ..., CN - boolean variables
Di - cost to go from source to vertex i
Pi - vertex preceding i in optimal path
s - source vertex

Algorithm:

Dijkstra (integer s)
  For i = 1 to N
    Ci = true;
    Cs = false;
  For i = 1 to N
    If Ci == true then
      Di = As,i;
      Pi = s;
  For i = 1 to N-1
    j = 1
    While Cj == false
      j = j + 1;
    Aux = ∞
    For k = j to N
      If Ck == true and Dk < Aux then
        v = k;
        Aux = Dk
    Cv = false;
    For k = 1 to N
      If Ck == true and Dk > Dv + Av,k then
        Dk = Dv + Av,k
        Pk = v;

```

Figure 10: Dijkstra's Algorithm

More information about Dijkstra's algorithm implementation can be found in Malpani and Chen (2002) and in Misra (2001).

To apply Dijkstra's algorithm successfully, edge costs must be calculated. The volume of the pipe that precedes each edge determines its cost. For example, the adjacency matrix for the vertices TK1, TK2, P1, P2, P5, and P6 shown in Fig. 8, is listed in table 2.

Table 2: Adjacency Matrix

To → From ↓	TK1	TK2	P1	P2	P5	P6
TK1	∞	∞	0	∞	∞	∞
TK2	∞	∞	∞	0	∞	∞
P1	P1Vol.	∞	∞	∞	P1Vol.	P1Vol.
P2	∞	P2Vol.	∞	∞	P2Vol.	P2Vol.
P5	∞	∞	P5Vol.	P5Vol.	∞	∞
P6	∞	∞	P6Vol.	P6Vol.	∞	∞

Table 2 shows that adjacency matrix elements must be loaded with ∞ (infinite, overflow, or a very large number) when there is no connection between the vertices. When there is connection, they must be loaded with the volume of the node corresponding to the origin spot. When origin is a tank, no volume is associated to it, and a zero appears in the matrix. When applying Dijkstra's algorithm, the minimum volume path will be found. Number of valve operations can be determined by the analysis of the status of the valves connected to the pipes in which product flows.

EVALUATION

A computer tool has been developed to create objects and search paths in the tank farm. It can display the tank farm layout, as well as the paths found by the algorithms described in the last session.

The first step is creating objects to represent the tanks, valves, pumps, and pipes as it is shown in Fig. 2. Once they are created, it is possible to search paths by selecting the source and destiny nodes. Since there are no pipelines or load facilities in the tank farm, the source and destiny nodes must be tanks.

Supposing that all the valves in the scenario displayed by Fig. 2 are closed, except V31, V4, VB1D, and VB1S (random initialization), a transfer operation between TK1 and TK2 was requested. Supposing that there is no difference among the product volumes inside the tanks, a pump is needed to perform product movement.

Using the path selection algorithm summarized in Fig. 9, twenty-one valid paths between TK1 and TK2 were found, and the best one is highlighted in Fig. 11. The solutions were found in 100 milliseconds, using an AMD Athlon® XP 2000+, and 256Mb DDR RAM.

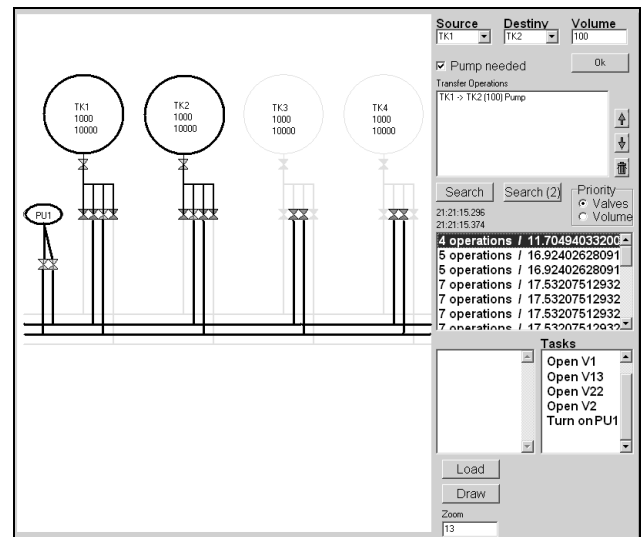


Figure 11: User-Interface

In Fig. 11, only the important pipes, valves, pumps, and tanks are displayed in foreground. Tasks needed to execute the transfer operation are informed by the application (opening valves V1, V13, V22, and V2, and turning on the pump PU1). Transfer costs are displayed in the center right of the application window (4 valve operations and a volume cost of 11.7).

The second best solution found by the algorithm requires five valve operations, and transfers the product through V1, V11, V31, V33, VB1S, VB1D, V22, and V2.

Operators can clearly see the path selections when the solutions are shown as an image, like Fig. 11. When showing the path, it is important to differentiate valves that must be opened, closed and the ones that must not be operated. In the interface, this differentiation is made by colors.

Dijkstra's algorithm found the same solution shown in Fig. 11, and took 55 milliseconds to find it, using an AMD Athlon® XP 2000+, and 256Mb DDR RAM.

CONCLUSIONS

A computer application to select paths in tank farms is a powerful tool to help the operational decision-making tasks on refinery environments. Today, in many refineries, men take decisions about transfer operations, and mistakes made in these decisions can cost a significant reduce of profitability. In some cases, incorrect valve operations can damage pipes making them leak and damage the environment.

Using the tool described in this paper, refinery operators' decisions about path selections can be simulated before being executed, and many possible mistakes in valve and pump operations can be warned and avoided.

It is possible to keep a history file that informs all transfer operations previously executed, so operators can check solutions adopted in the past.

The tool also can be used for training new operators, who will be able to evaluate many transfer possibilities in the pipe network, simulate transfer operations, and learn about path selection decisions before making them in the real tank farm.

Although only two search methods were described, the modeling shown in this paper can reproduce efficiently a tank farm, its components, and connections. This modeling allows representing wide tank farms efficiently as graphs, and many other methods for path selection can be applied and tested, like the algorithms described in Saunders and Takaoka (2003), or even solutions implemented in hardware, like Hopfield neural networks (Bousño-Calzón and Figueiras-Vidal, 1997).

ACKNOWLEDGEMENTS

The authors acknowledge the financial support of Brazilian National Agency of Petroleum (PRH-ANP/MCT 10 CEFET-PR).

REFERENCES

- Boroujerdi, A., Uhlmann, J. 1998. "An efficient algorithm for computing least cost paths with turn constraints" *Information Processing Letters* 67, 317-321.
- Bousño-Calzón, C., Figueiras-Vidal, A. R. 1997. "A bank of Hopfield neural networks for the shortest path problem" *Signal Processing* 61, 157-170.
- Dijkstra, E. W. 1959. "A note on two problems in connection to graphs" *Numer. Math.* 1, 83-89.
- Korkmaz, T. and Krunz, M. 2001. "A randomized algorithm for finding a path subject to multiple QoS requirements" *Computer Networks* 36, 251-268.
- Kuttelwascher, H. and Rhemann, H. 1995. "Tank farm automation with DCS" *Hydrocarbon Processing*, Nov, 149-154.
- Malpani, N. and Chen, J. 2002. "A note on practical construction of maximum bandwidth paths" *Information Processing Letters* 83, 175-180.
- McCormack, J. E. and Roberts, S. A. 1996. "Exploiting object oriented methods for multi-modal trip planning systems" *Information and Software Technology* 38, 409-417.
- Mikler, A. R., Wong, J. S. K., Honavar, V. 1998. "An Object Oriented Approach to Simulate Large Communication Networks" *J. Systems Software* 40, 151-164.
- Misra, J. 2001. "A walk over the shortest path: Dijkstra Algorithm viewed as fixed-point computation". *Information Processing Letters* 77, 197-200.
- Pinto, J. M., Joly, M., Moro, L. F. L. 2000. "Planning and scheduling models for refinery operations" *Computers and Chemical Engineering* 24, 2259-2276.
- Saunders, S., Takaoka, T. 2003. "Improved shortest path algorithms for nearly acyclic graphs" *Theoretical Computer Science* 293, 535-556.
- Solka, J. L., Perry, J. C., Poellinger, B. R., Rogers, G. W. 1995. "Fast computation of optimal paths using a parallel Dijkstra algorithm with embedded constraints" *Neurocomputing* 8, 195-212.
- Takahashi, K., Kateeshock, T. 1992. "Expert Systems for Refinery Off-Site Facility Management" *ISA Transactions* 31, No. 2, 67-75.
- Wöllhaf, K., Engell, S. 1995. "Object-oriented modeling and simulation of batch plants" *Mathematics and Computers in Simulation* 39, 519-525.

SIMULATION IN ROBOTICS AND MULTIBODY APPLICATIONS

Robot Vehicles Planning and Tracking

GEOMETRIC PARALLEL PARKING PLANNER FOR CAR-LIKE VEHICLES

Luis Gracia and Josep Tornero
Dpto. Ingeniería de Sistemas y Automática
Universidad Politécnica de Valencia
46022 Valencia (Spain)
Email: {luigraca,jtornero}@isa.upv.es

KEYWORDS

Autonomous Robots, Collision Avoidance, Trajectory Generation.

ABSTRACT

The paper presents a parallel parking planner (trajectory generator) for car-like vehicles, based on geometric collision avoidance calculations. Important improvements have been introduced with respect to other geometrical approaches. A simulation environment has been design for testing the parking planner under different situations and vehicles. Several simulations are includes for validate the planners and the simulation software.

INTRODUCTION

The automation of industrial processes is frequently based on the use of autonomous vehicles (wheeled mobile robots). Usually, they are involved in movements such as docking and parallel parking with limited space. This issue becomes complex when classical car-like vehicles are considered, due to its nonholonomic constraints (Laumond 1992).

Many authors have tried to solve this kind of situations giving efficient algorithms for generating forward and backward movements (Barraquand and Latombe 1989).

In fact, parallel parking is an interesting problem, which justify the design of powerful planers, some of them based on geometrical reasoning or optimal criteria.

Next section describes possible planners for parking purpose, meanwhile the third section develops one of them with important improvements with respect to other geometrical approaches. Finally, fourth section shows planning results in a simulation environment.

ALTERNATIVE PLANNERS FOR PARKING

General Purpose Planners

General planners can be used for planning parallel parking based on different approaches. Some planners are based on tree graphs, where the leafs are postures (position + orientation) of the vehicle and the branches are the paths from a posture to another. Then, the planner checks, during the construction of the tree graph, if the goal has been achieved. See for instance (Latombe 1991), (Laumond et al. 1994), etc. Their main disadvantage is the high

computational cost required, what disables it for real time applications.

Other planners use more particular methods based on tools like fuzzy, linearization by state feedback, neural networks, reactive architecture, etc. As an example, (Tilbury et al. 1993) uses sinusoids for steering a self-guided vehicle with n-trailers in a parallel parking.

Specific Planners for Parallel Parking

The planners presented in (Paromtchik and Laugier 1996) [1] and (Jiang et al. 1998) [2] are specific for accomplishing parking maneuvers in three phases: 1) Detection with ultrasonic sensors of the parking space (rectangle); 2) Positioning to the edge of the parking space; 3) Maneuvering to the parking position with a unified pattern.

In [2], maneuvers are calculated from geometric characteristics, meanwhile [1] applies pre-established control actions, which would produce appropriated maneuvers. The algorithm shown in [2] needs to stop the vehicle in the middle of the maneuver in order to reorient the steering wheels, meanwhile in [1] the reorientation is progressive. Finally, [1] takes more time in computation, since it needs numeric integration, meanwhile in [2] all is computed analytically.

In this paper, we present a geometric planner, which improves the approach given in [2] in two aspects: 1) Collision avoidance with the front and the back obstacles, not just with the lateral one; 2) Optimization of the positioning point in second phase.

PARALLEL PARKING PLANNER

General Characteristics of the Planner

The planner developed is based on geometric calculations using obstacles collision avoidance in the three traditional phases as seen in Figure 1.

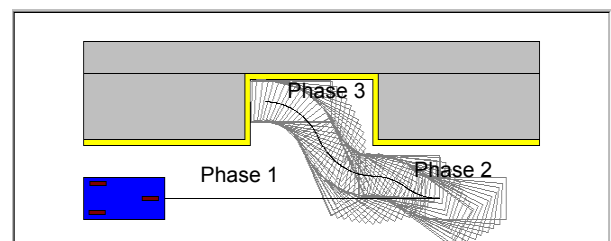


Figure 1: Parking Phases

The phases are:

1) Detect the parking space (rectangle) passing by.

2) Maneuver back to the positioning point. This point is optimized, on the prolongation of the edge of the front obstacle, to be as close as possible to the front obstacle but avoiding collisions in the first backward maneuver.

3) Maneuver, backward and forward, until the vehicle achieves the goal, avoiding collisions with lateral, front and back obstacle. Each maneuver has two circular paths, where the equivalent steering wheel travels from one end to the opposite. The vehicle is parallel to the parking space at the beginning and at the end of each maneuver.

Geometric Characterization for Parking

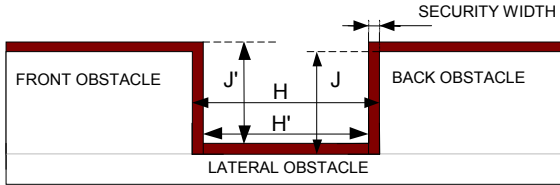


Figure 2: Geometric Parameters of the Parking Space

Figure 2 shows the geometric parameters of the parking space, where J: width of the parking space; H: length of the parking space; MS: security margin; J': effective width of the parking space, equivalent to J; H': effective length of the parking space, equivalent to (H-2·MS).

Figure 3 shows the geometric parameters of a car-like vehicle where the equivalent steering wheel has been depicted. If more than one wheel is present, the Ackerman mechanism guarantee that equivalence.

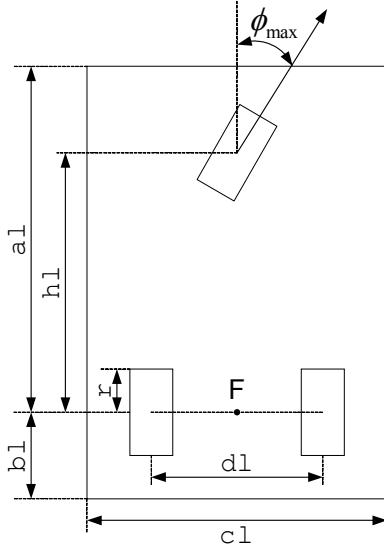


Figure 3: Geometric Parameters of the Car-like Vehicle

In this Figure al: distance between the rear axle and the front of the vehicle; bl: distance between the rear axle and the back of the vehicle; hl: distance between the rear axle and the steering wheel; dl: distance between the centers of the fixed wheels of the rear axle; cl: width of the vehicle; r: radius of the wheels; ϕ_{\max} : maximum orientation of the equivalent steering wheel; F: point in the middle of the rear axle that tracks the trajectory.

Now, some useful geometric relationships are shown from Figure 4, 5 and 6:

→ Relationship between the turning radius ρ (distance from F to the instantaneous center of rotation, ICR) and the orientation of the equivalent steering wheel:

$$\rho = \frac{hl}{\tan(\phi)} \quad (1)$$

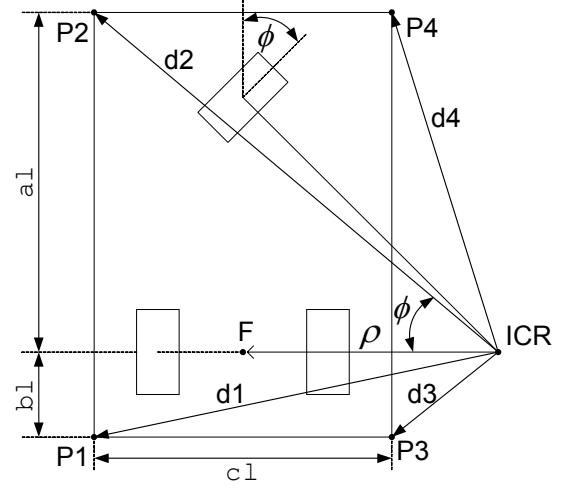


Figure 4: Description of ICR, d_1 , d_2 , d_3 and d_4

→ Relationship between the turning radius ρ and the distance d_i from the ICR to the corner P_i of the vehicle:

$$\begin{aligned} d_1 &= \sqrt{bl^2 + \left(\rho + \frac{cl}{2}\right)^2} & d_2 &= \sqrt{al^2 + \left(\rho + \frac{cl}{2}\right)^2} \\ d_3 &= \sqrt{bl^2 + \left(\rho - \frac{cl}{2}\right)^2} & d_4 &= \sqrt{al^2 + \left(\rho - \frac{cl}{2}\right)^2} \end{aligned} \quad (2)$$

→ Relationship between the orientation of the vehicle θ and the position (x_i, y_i) of the corner P_i in Figure 5:

$$(x_{P_i}, y_{P_i}) = (d_i \cdot \cos(\theta - \delta_i), d_i \cdot \sin(\theta - \delta_i)) \quad (3)$$

$$(x_{P_j}, y_{P_j}) = (d_j \cdot \cos(\theta + \delta_j), d_j \cdot \sin(\theta + \delta_j))$$

where $i = 1, 2, j = 2, 3$ and

$$\delta_1 = \arctan\left(\frac{bl}{\rho + cl/2}\right) \quad \delta_2 = \arctan\left(\frac{al}{\rho + cl/2}\right)$$

$$\delta_3 = \arctan\left(\frac{bl}{\rho - cl/2}\right) \quad \delta_4 = \arctan\left(\frac{al}{\rho - cl/2}\right)$$

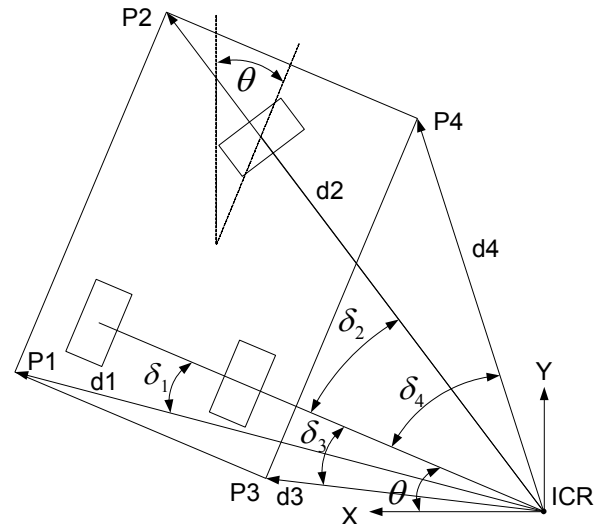


Figure 5: Relationship between θ and the Position of P_i

→ Longitudinal and transversal displacements in a path consist of two symmetric arcs (see Figure 6) of value ξ :

$$X = 2 \cdot \rho \cdot (1 - \cos(\xi)), Y = 2 \cdot \rho \cdot \sin(\xi), Z = \rho \cdot (1 - 2 \cdot \cos(\xi)) \quad (4)$$

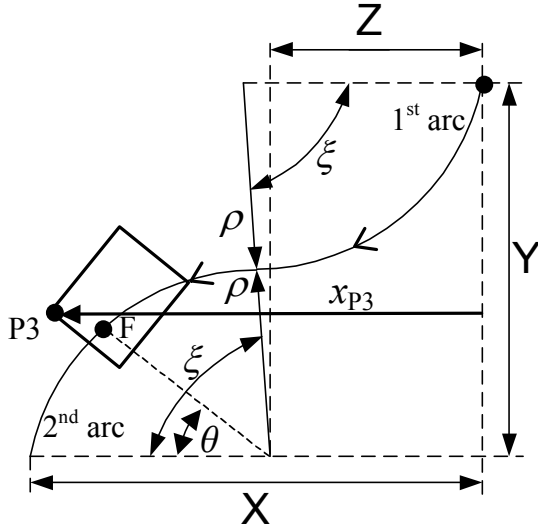


Fig. 6: Displacements for Two Symmetric Arcs of Value ξ

Characterization of Collisions

Possible collisions are with the lateral, front and back obstacles when maneuvering and with the front obstacle in the first maneuver when departing positioning point.

Lateral collision is always produced with one of the nearest corner of the vehicle to the lateral obstacle in the direction of the movement (e.g. in Figure 4, P1 may collide in backward maneuver or P2 in forward maneuver). Then, in order to avoid lateral collision, in a maneuver, no collision of the corresponding corner must be guaranteed.

In Figure 6, it is shown the two symmetric arcs that conform an approaching maneuver. It is enough to analyze the second arc, which is closer to the lateral obstacle. Then, the lateral distance has to be computed for checking the lateral space available.

From equations (3) and (4), it is possible to compute the lateral distance between the corner P1 or P2 (movement backward or forward) depending on the angle θ of the second arc,

$$X_{Pi}(\theta) = di \cdot \cos(\theta - \delta_i) + \rho \cdot (1 - 2 \cdot \cos(\xi)) \quad \text{where } i=1,2 \quad (5)$$

with a maximum in:

$$\hat{X}_{Pi} = di + \rho \cdot (1 - 2 \cdot \cos(\xi)) \quad \text{for } \hat{\theta}_{Pi} = \delta_i \quad (6)$$

Lateral collision detection is checked by comparing (6b) to the lateral space available.

Checking collision implies to verify that the critical corner never collides.

Front / back collision is produced with the nearest corner of the vehicle in the direction of the movement. That is P2/P3 in backward maneuver and P1/P4 in a forward maneuver.

The back collision in the backward maneuver, involves P3 and only the second arc, and is produced if the local minimum of y_{P3} in (3a) is for a value $\hat{\theta} \in]0, \xi[$, where:

$$\hat{\theta}_3 = \delta_3 - \frac{\pi}{2} \quad \text{with } 0 \leq \theta \leq \xi, \quad 0 \leq \delta_3 \leq \pi \quad (7)$$

Then, the singular point (7) is outside the range of θ when:

$$\rho \geq cl/2 \quad \text{or} \quad \rho \leq cl/2 - bl \cdot \tan(\xi) \quad (8)$$

In back and front collisions, other corners can be analyzed giving similar results (for P2 and P4 bl is replaced with al).

The following algorithm is proposed:

```

if (8a) → no back/front collision
else if  $\rho \leq cl/2 - ab \cdot \tan(\xi)$  (9)
    {  $ab=bl$  for P1,P3 and  $ab=al$  for P2,P4 }
    → no back/front collision
else → back/front collision

```

Algorithm 1: Detection of Back/Front Collision

In addition, *collisions at first maneuver* (between the front obstacle and corner P2) are only produced at the second arc.

In order to avoid these collisions, the minimum for $dx1$ in Figure 7 is calculated with an algorithm, in such a way that P2 touches the origin of the system (X,Y). For this calculation, expression (10), obtained from (3) and (4), is used.

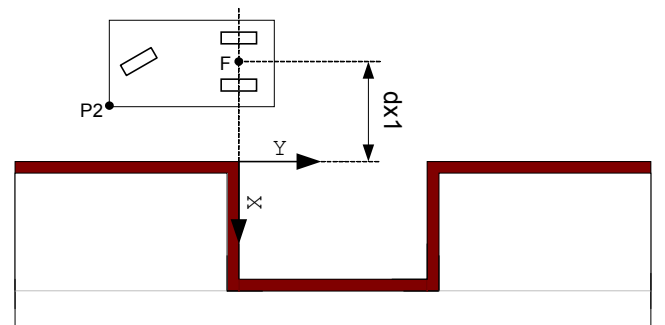


Figure 7: Characteristics of First Maneuver Collision

$$y_{P2}(\theta) = 2 \cdot \rho \cdot \sin(\xi) - \left(\rho \cdot \sin(\theta) + \frac{cl}{2} \cdot \sin(\theta) + al \cdot \cos(\theta) \right) \quad (10)$$

$$x_{P2}(\theta) = \rho \cdot (1 - 2 \cdot \cos(\xi)) + \rho \cdot \cos(\theta) + \frac{cl}{2} \cdot \cos(\theta) - al \cdot \sin(\theta) - dx1$$

In addition, possible lateral and back collisions has to be considered in the algorithm for the definitive value of $dx1$.

Once we state the distance $dx1$, which characterizes the positioning point, it is necessary to calculate the minimum distance $dy3$ of Figure 8 for stopping the first straight movement of the vehicle.

This calculation is carried out by an algorithm, with the inputs $dx1$ and $dx2$ (initial vehicle separation), that takes into account lateral collision avoidance (using (6)) with the front obstacle in the second phase.

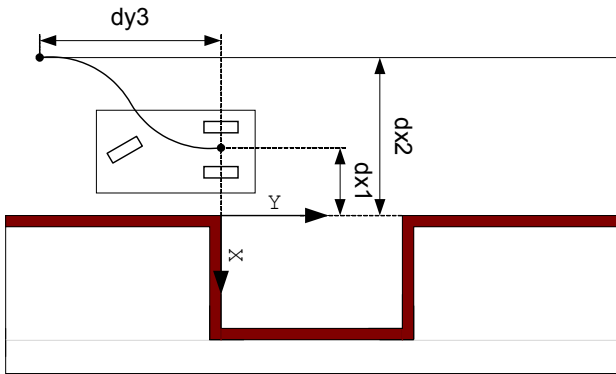


Figure 8: Previous Maneuver to de Positioning Point

Finally, in order to characterize each necessary maneuver of the third phase an algorithm has been developed. It uses the actual distance available to the lateral obstacle as an input and calculates the values for ξ and ρ in that maneuver. For this, it takes into account lateral, front and back collision avoidance using (6) and Algorithm 1.

The repetitions of last algorithm stop when the lateral distance, between the vehicle and the lateral obstacle, is less than a desired final separation.

PLANNER RESULTS IN SIMULATION

In order to validate the developed planner, it has been implemented a simulation environment. An interface has been also created for introducing the geometric characteristics of the vehicle as seen in Figure 9 and the parking space (Figure 10). Even dynamics values of the vehicle may be specified for monitoring a simulated tracking.

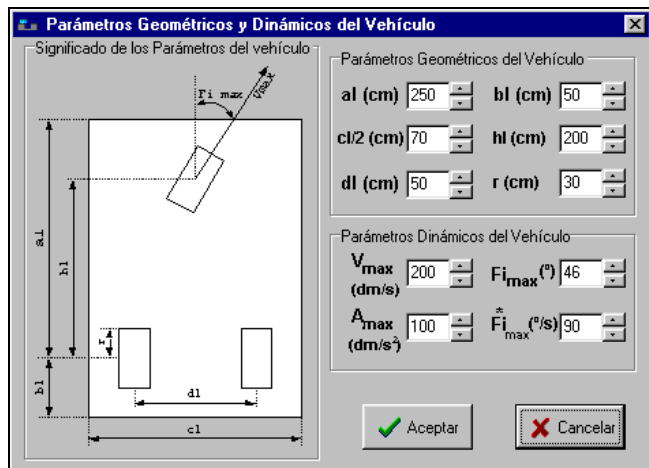


Figure 9: Vehicle Parameters Interface

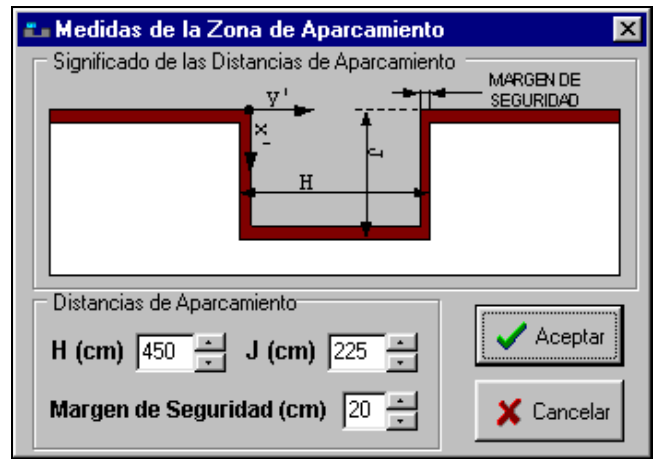


Figure 10: Parking Space Parameters Interface

In Figure 11, it is shown a parking example for the following parameters:

Parameter	Value	Parameter	Value
al	150 cm	H	335 cm
bl	50 cm	J	200 cm
cl	140 cm	MS	20 cm
hl	105 cm	dx2	70 cm
dl	50 cm	Final separation	7 cm
ϕ_{max}	46°		

The computation time results 0.438 milliseconds and two maneuvers are required.

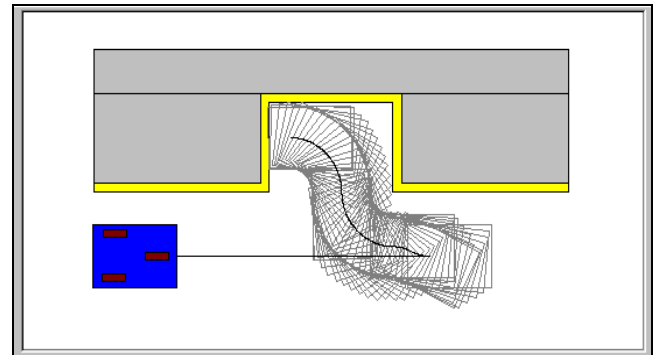


Figure 11: Parking Example 1 with the Planner Developed

Also, Figure12 and 13 show another example for the following parking parameters:

Parameter	Value	Parameter	Value
al	75 cm	H	330 cm
bl	15 cm	J	360 cm
cl	350 cm	MS	12 cm
hl	60 cm	dx2	17 cm
dl	130 cm	Final separation	17 cm
ϕ_{max}	46°		

Note that, in this case it has been considered a much more wide vehicle. This would correspond, for instance, to the case of an agronomic vehicle.

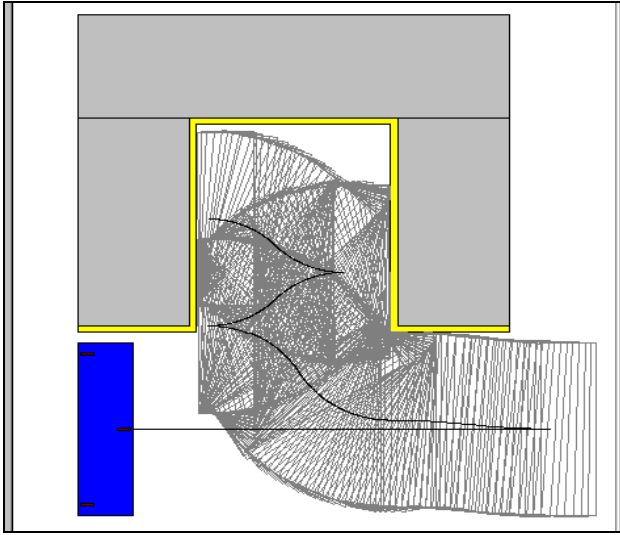


Figure 12: Parking Example 2 with the Planner Developed

In this case, computation time is 1.3 milliseconds and four maneuvers are required.

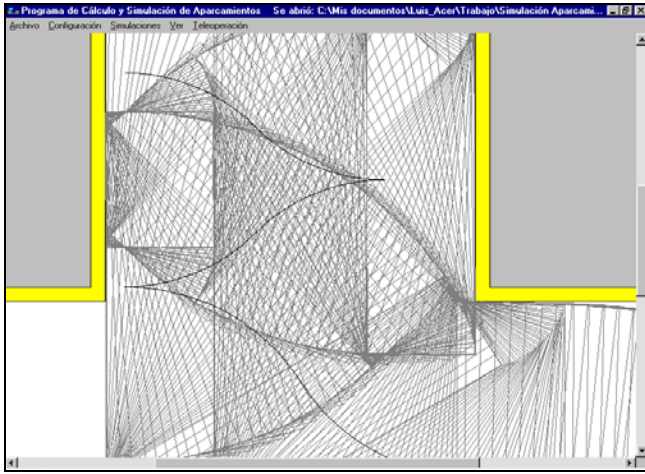


Figure 13: Zoom of the Parking Example 2

Figure 14 shows number of maneuvers required respect to the longitudinal space available in the terms of the ratio given in (11). Note that the maneuvers required grow rapidly when ratio goes below 30%.

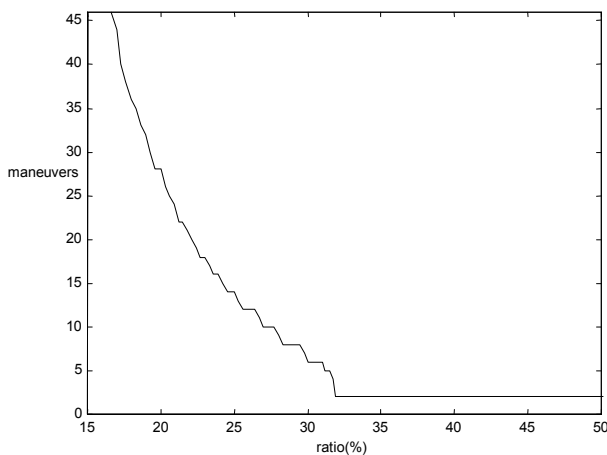


Fig. 14: Maneuvers – Ratio (%) Relationship in Example 1

$$\text{ratio}(\%) = 100 \cdot \frac{\text{available space} - \text{vehicle length}}{\text{available space}} \quad (11)$$

$$\text{ratio}(\%) = 100 \cdot \frac{(H - 2 \cdot MS) - (a_l + b_l)}{(H - 2 \cdot MS)}$$

It is interesting to remark that planner has been tested in simulation but can also be implemented on a real vehicle.

CONCLUSIONS

A parallel parking planner for car-like vehicles, based on geometric collision avoidance calculations has been implemented. Important improvements have been introduced with respect to other geometrical approaches in the line of vehicle collision avoidance with front and back obstacles and optimization criteria. The low computation time required (around few millisecond) makes the planner appropriate for real time applications.

In addition, a simulation environment has been design for testing this parking planner under different situations and vehicles. Several simulations are includes for validate the planners and the simulation software, indicating how the number of maneuvers increases in function of the space available.

Acknowledgments

This research has been partially funded by the Spanish Government and FEDER. Research Projects DPI2000-0362-P4-05 and DPI2001-2689-C03-02.

REFERENCES

- Barraquand J. and J.-C. Latombe. 1989. "On non-holonomic Mobile Robots and Optimal Maneuvering", *Revue d'Intelligence Artificielle*, vol. 3, no. 2, pp. 77-103.
- Gracia, L. and J. Tornero. 2002. "Kinematic Control System for Car-Like Vehicles" *Advances in Artificial Intelligence – Iberamia 2002*, 882-892.
- Jiang K., D. Z. Zhang and L. D. Seneviratne. 1998. "A Parallel Parking System for a Car-like Robot with Sensor Guidance", *Proc. Instn. Mech. Engrs.*, 1998, Vol. 213 Part C.
- Latombe J.-C. 1991. "Robot Motion Planning", *Kluwer Academic Publishers*, chapter 7.
- Laumond J.-P. 1992. "Singularities and Topological Aspects in Nonholonomic Motion Planning" in *Nonholonomic Motion Planning*, Eds. New York: The Kluwer International Series in Engineering and Computer Science 192.
- Laumond J.-P., P. E. Jacobs, et al. 1994. "A Motion Planner for Nonholonomic Mobile Robots", *IEEE Transactions on Robotics and Automation*, 1994, Vol. 10 No. 5.
- Paromtchik I. E. and C. Laugier. 1996. "Motion Generation and Control for Parking an Autonomous Vehicle", *Proc. Of The IEEE Int. Conf. on Robotics and Automation*, pp. 3117-3122.
- Tilbury D., R. Murray and S. Sastry. 1993. "Trajectory Generation for the N-Trailer Problem Using Goursat Normal Form", *Proceedings of the IEEE Conference on Decision and Control*.

MULTI-RATE LINE TRACKING FOR MOBILE ROBOTS BASED ON ARTIFICIAL VISION AND ODOMETRY

Leopoldo Armesto, Luis Gracia & Josep Tornero
Dpto. Ingeniería de Sistemas y Automática
Universidad Politécnica de Valencia
46022 Valencia (Spain)
Email: {leoaran,luigraca,jtornero}@isa.upv.es

KEYWORDS

Line-Tracking, Multi-Rate Sampling, Kinematic Control, Vision.

ABSTRACT

In this paper, the problem of line-tracking for mobile robots using artificial vision have been presented based on a multi-rate structure. Two different multi-rate control architectures have been implemented and tested: The first control architecture estimates the robot posture from the odometry (angular velocities of the wheels) using the camera data when available. The second control architecture uses a multi-rate Extended Kalman Filter (MR-EKF) to fusion sensor data (image and odometers). The results of both multi-rate algorithm are analyzed and compared. Both multi-rate algorithms give similar results except when the noise is applied to the measurements. In this case, the MR-EKF gives better performances because it considers a multi-rate systems dynamic model.

INTRODUCTION

The automation of industrial processes is frequently based on the use of wheeled mobile robots. Usually, they are involved in path tracking and sometimes use vision systems (Marchant 1995), (Gracia and Tornero 2002-1), (Gracia and Tornero 2002-2).

In the those applications, sampling time for acquiring and processing vision data becomes critic, too high for the kinematic control stability. Given that image processing usually requires a high computation cost, the robot behavior worsens considerably for a kinematic control running at the sampling frequency of the image.

On the contrary, odometer sensors can improve robot behaviour, due to the fact that they can be easily measured at high sampling frequency.

In this paper, two different multi-rate algorithms are proposed for the line tracking problem, by combining data from the odometry and the vision system. In the first algorithm, the robot position is estimated using the vision system only when the information is available while the odometry is the only information used when there is not image. In the second algorithm, a Multi-rate Extended Kalman Filter is used for the robot position estimation.

The paper is organized as follows: The line-tracking based on a vision system algorithm is firstly described, where the problematic of the robot positioning with respect to the line and the kinematic control are considered; secondly, it is

studied the robot behavior for low-frequency sampling rates; finally the two multi-rate algorithms are analyzed for noise robustness. Conclusions are given at the end of the paper.

LINE TRACKING WITH A VISION SYSTEM

The first step in order to perform line tracking is to position the vehicle with respect to the reference line and the second step is to carry out control reducing position error. These two basic issues are described next.

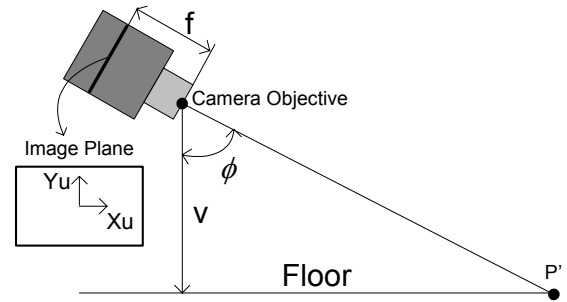


Figure 1: Vision System. Image Processing

Positioning with a Vision System

In the first step, the reference line from the image plane of the camera are detected by analyzing the image and applying the Hough transform (Gracia and Tornero 2002-1). This process requires much time in computation.

Then, using a particularization of the camera model (Gracia and Tornero 2002-1), we get the position of the robot with respect to the line (h, ψ) (see figure 2) as follows:

$$\psi = \arctan \left(\cos(\phi) \cdot \frac{(x_{u1} - x_{u2})}{(y_{u2} - y_{u1})} + \frac{\sin(\phi)}{f} \cdot \frac{(y_{u2} \cdot x_{u1} - y_{u1} \cdot x_{u2})}{(y_{u1} - y_{u2})} \right)$$

$$h = \frac{v}{f} \cdot \frac{\cos(\psi)}{\cos(\phi)} \cdot \frac{(y_{u1} \cdot x_{u2} - y_{u2} \cdot x_{u1})}{(y_{u1} - y_{u2})}$$

where (x_{ui}, y_{ui}) are the projections of line points on the image plane referred to the coordinate system (X_u, Y_u) , centered on the image plane; f is the focal distance of the camera; v is the altitude (constant) of the camera objective with respect to the reference line plane; ϕ is the pitch angle (constant) of the camera, ψ is the roll angle of the camera with respect to the reference line, h is the distance between the tracking point P' and the line.

Kinematic Control for Line Tracking

Further developments assume a vehicle with differential wheels. If exist any additional wheel, its orientation must be coordinate with the differential mechanism.

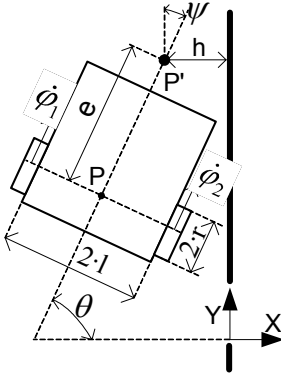


Figure 2: Vehicle with differential wheels

The above figure shows the scheme of the typical differential vehicle, with the following variables and parameters: P: midpoint of the fixed wheels axle; (X, Y): world coordinate system, on the tracking line; (x, y): position of the point P with respect to the world coordinate system; θ : vehicle orientation with respect to the world coordinate system; (x, y, θ): vehicle posture; P': point attached to the WMR that tracks the reference (typically the cross point between the camera axel and the floor); e: distance between P and P'; r: radius of the wheels; 2l: fixed wheels separation; $\dot{\phi}_1, \dot{\phi}_2$: rotation velocities of the fixed wheels; (h, ψ): position parameters with respect to the line, obtained from the vision system.

The (direct) kinematic model of the vehicle, according to Gracia and Tornero 2001, is:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \frac{r}{2 \cdot l} \begin{pmatrix} l \cdot \cos(\theta) & l \cdot \cos(\theta) \\ l \cdot \sin(\theta) & l \cdot \sin(\theta) \\ -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \end{pmatrix} \quad (1)$$

while, taking into account the kinematic control proposed in Gracia and Tornero 2002-2 for line tracking, and considering the wheels dynamics and sensing, the following scheme is obtained:

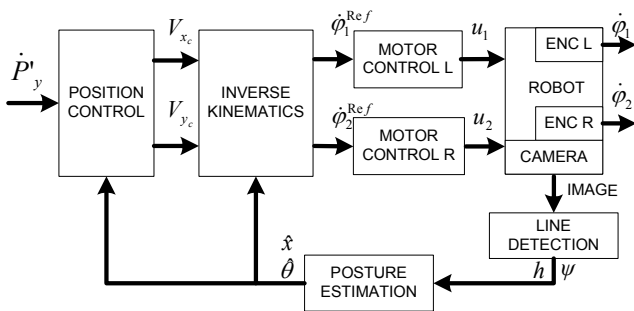


Figure 3: Line-Tracking Control Architecture

Note that the previous kinematic control architecture includes the following blocks: Position-Control, Inverse-Kinematics and Posture-Estimation, implemented respectively by the matrix equations:

$$\begin{pmatrix} V_{x_c} \\ V_{y_c} \end{pmatrix} = \begin{pmatrix} -a_x \cdot (x + e \cdot \cos(\theta)) \\ \dot{P}'_{y_ref} \end{pmatrix}$$

$$\begin{pmatrix} \dot{\phi}_1^{ref} \\ \dot{\phi}_2^{ref} \end{pmatrix} = \frac{1}{r \cdot e} \cdot \begin{pmatrix} l \cdot \sin(\theta) & e \cdot \sin(\theta) \\ +e \cdot \cos(\theta) & -l \cdot \cos(\theta) \\ -l \cdot \sin(\theta) & e \cdot \sin(\theta) \\ +e \cdot \cos(\theta) & +l \cdot \cos(\theta) \end{pmatrix} \cdot \begin{pmatrix} V_{x_c} \\ V_{y_c} \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} = \begin{pmatrix} -(e \cdot \sin(\psi) + h) \\ \text{Indetermined} \\ \frac{\pi}{2} - \psi \end{pmatrix} \quad (2)$$

At this point, an example of a mobile robot tracking a line is showed in order to clarify concepts. From the image processing, the line parameters h and ψ are calculated at a sampling rate of $T = 0.05 \text{ sec}$, where from (2), the posture estimation algorithm is in discrete-time as follows:

$$\hat{x}(k) = -h(k) - e \cdot \sin(\psi(k))$$

$$\hat{\theta}(k) = \frac{\pi}{2} - \psi(k)$$

The mobile robot half separation between wheels is $l = 0.1715 \text{ m}$; meanwhile the controller parameters are $a_x = 0.8 \text{ s}^{-1}$, $\dot{P}'_y = 1 \text{ m/s}$ and $e = 0.39 \text{ m}$. The speed control for each wheel is supposed to behave like a first order model with time constant of $\tau = 0.2 \text{ sec}$.

Figure 4, shows the robot's track for a line situated perpendicular to the initial robot position at a distance of 4m, that is $x(0) = -4 \text{ m}$ and $\theta(0) = 0^\circ$. As it can be seen the robot converges exponentially to the line.

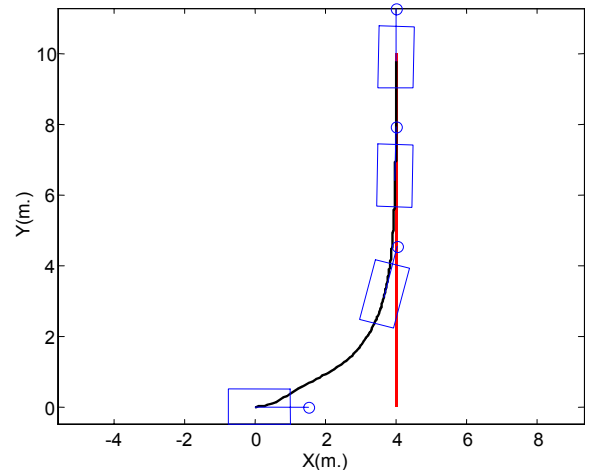


Figure 4: Line-Tracking Example With $T = 0.05 \text{ sec}$

However, the image processing requires a high-computational cost, where for instance, with a 800 Mhz Celeron processor it takes around $T = 0.25 \text{ sec}$ in computing the line parameters. The robot behaviour worsens drastically for lower frequency sampling rates, as depicted in figure 5.

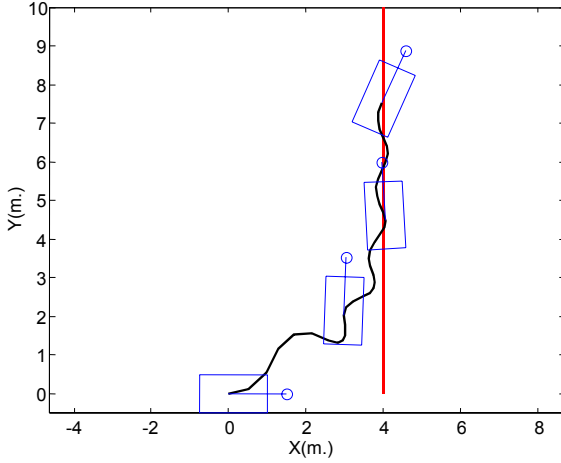


Figure 5: Line-tracking example with $T = 0.25$ sec

It is interesting to remark that robot behavior is in fact a combination of two effects: from one side, the reduction of the sampling frequency and on the other side, the wheels dynamics of the speed control.

MULTI-RATE LINE TRACKING

In order to improve the system response, two different multi-rate control architectures are proposed in this section. The first control architecture estimates the robot posture by combining the posture obtained from line parameters at a low-frequency sampling rate of $T = 0.25$ sec with the posture computed from the robot odometry (angular velocities of the wheels) at a high-frequency sampling rate of $T = 0.05$ sec. The second control architecture uses a multi-rate Extended Kalman Filter (Sorenson, 1985) (MR-EKF) to fusion sensor data (image and odometers). The results of both multi-rate algorithm are compared at the end of this section.

Based on the ideas of (Tornero, 1985) and (Tornero, *et. al.* 1999), it is defined a multi-rate sampling matrix as follows:

$$\Delta(k) = \begin{cases} I & \text{iff } \text{Mod}(k, N) \\ 0 & \text{otherwise} \end{cases}$$

where k is the discrete-time index of sampling instants $t = k \cdot T$ and N is the relation between the low-sampling period and the high-one, $\bar{T} = N \cdot T$. Note, that we will use $\Delta(k)$ for defining the camera's samplings, where $\Delta(k) = I$ indicates that a sampling has been produced.

Multi-Rate Posture Estimation

The multi-rate posture estimation, parts from a simple but at the same time coherent idea: the robot posture is computed using (2) when the image data is available otherwise it is computed from the direct kinematic model (1) using the wheels' velocities (always available at high sampling frequency).

Hence, the control architecture for the line tracking, is depicted in figure 6, where thick lines represent signals computed at a high-frequency sampling rate, meanwhile thin lines are at a low-one.

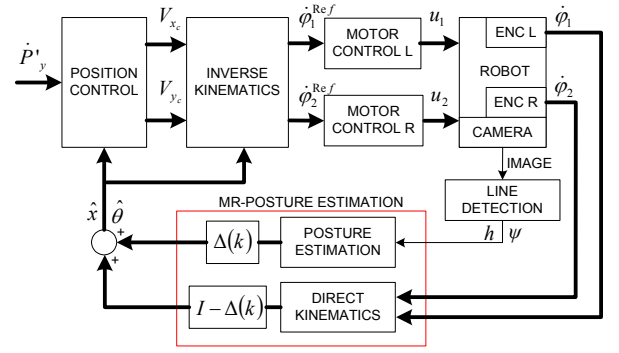


Figure 6: Control Architecture of MR-Posture Estimation

In this sense, the multi-rate posture estimation is formulated as follows:

$$\begin{bmatrix} \hat{x}(k+1) \\ \hat{\theta}(k+1) \end{bmatrix} = \Delta(k) \cdot \begin{bmatrix} h(k) + e \cdot \sin \psi(k) \\ \frac{\pi}{2} - \psi(k) \end{bmatrix} + [I - \Delta(k)] \cdot \begin{bmatrix} \hat{x}(k) \\ \hat{\theta}(k) \end{bmatrix} + \begin{bmatrix} \frac{r}{2} \cdot \cos(\hat{\theta}(k)) \cdot T & \frac{r}{2} \cdot \cos(\hat{\theta}(k)) \cdot T \\ -\frac{r}{2 \cdot l} \cdot T & \frac{r}{2 \cdot l} \cdot T \end{bmatrix} \cdot \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix}$$

Based on the example presented above, the multi-rate scheme has been implemented, obtaining similar results as in the single-rate case (where the line parameters were supposed to be computed at a high-frequency).

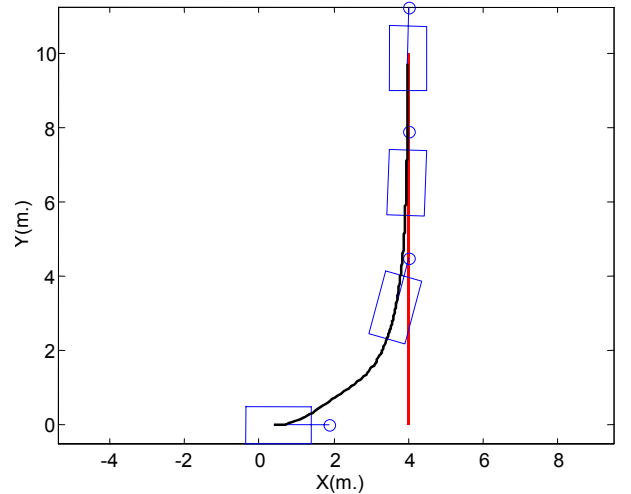


Figure 7: Line-Tracking With MR-Posture Estimation

Multi-Rate Extended Kalman Filter

A multi-rate Extended Kalman Filter has been also implemented in order to compare both multi-rate algorithms. The EKF (Tornero, *et. al.*, 2001) is based on a second order kinematic model for the mobile robot defined as follows,

$$\begin{cases} x_{k+1} = f(x_k) \cdot x_k + g(x_k) \cdot u_k + w_k \\ y_k = h(x_k) + v_k \end{cases}$$

being, $x_k = [x(k) \ v_x(k) \ y(k) \ v_y(k) \ \theta(k) \ \omega(k)]^T$ the states vector, $u_k = [\dot{\phi}_1(k) \ \dot{\phi}_2(k)]^T$ the input command (provided by odometry) and $y_k = [h(k) \ \psi(k)]^T$ the line parameters obtained from the image processing. The noise

input w_k and the noise output v_k are supposed to be white noises with no bias. The remainder system matrices are defined as follows,

$$f(x_k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$g(x_k) = \begin{bmatrix} \frac{r}{2} \cdot \cos(\theta(k)) \cdot T & \frac{r}{2} \cdot \cos(\theta(k)) \cdot T \\ \frac{r}{2} \cdot \cos(\theta(k)) & \frac{r}{2} \cdot \cos(\theta(k)) \\ \frac{r}{2} \cdot \sin(\theta(k)) \cdot T & \frac{r}{2} \cdot \sin(\theta(k)) \cdot T \\ \frac{r}{2} \cdot \sin(\theta(k)) & \frac{r}{2} \cdot \sin(\theta(k)) \\ -\frac{r}{2 \cdot l} \cdot T & \frac{r}{2 \cdot l} \cdot T \\ -\frac{r}{2 \cdot l} & \frac{r}{2 \cdot l} \end{bmatrix}$$

$$h(x_k) = \begin{bmatrix} -(x(k) + e \cdot \cos(\theta(k))) \\ \frac{\pi}{2} - \theta(k) \end{bmatrix}$$

To compute the EKF, it is required to compute the first order Taylor approximations of $f(x_k)$, $g(x_k)$ and $h(x_k)$:

$$F_k = \left. \frac{\partial f(x_k)}{\partial x_k} \right|_{x_k = \hat{x}_{k/k}} \quad G_k = \left. \frac{\partial g(x_k)}{\partial x_k} \right|_{x_k = \hat{x}_{k/k}} \quad H_k = \left. \frac{\partial h(x_k)}{\partial x_k} \right|_{x_k = \hat{x}_{k/k}}$$

Then, the filter is computed by applying the following recursive equations:

$$\begin{aligned} \hat{x}_{k+1/k} &= f_k(\hat{x}_{k/k}) + g_k(\hat{x}_{k/k}) \cdot u_k \\ P_{k+1/k} &= F_k \cdot P_{k/k} \cdot F_k^T + G_k \cdot Q_k \cdot G_k^T \\ S_{k+1} &= H_k \cdot P_{k+1/k} \cdot H_k^T + R_k \\ K_k &= P_{k+1/k} \cdot H_k^T \cdot S_{k+1}^{-1} \\ \hat{x}_{k+1/k+1} &= \hat{x}_{k+1/k} + K_k \cdot \Delta(k) \cdot [y_k - h_k(\hat{x}_{k+1/k})] \\ P_{k+1/k+1} &= P_{k+1/k} - K_k \cdot H_k \cdot P_{k+1/k} \end{aligned}$$

where Q_k y R_k are the covariance matrices of input and output noises, respectively.

The control architecture used for the MR-EKF is represented by figure 8:

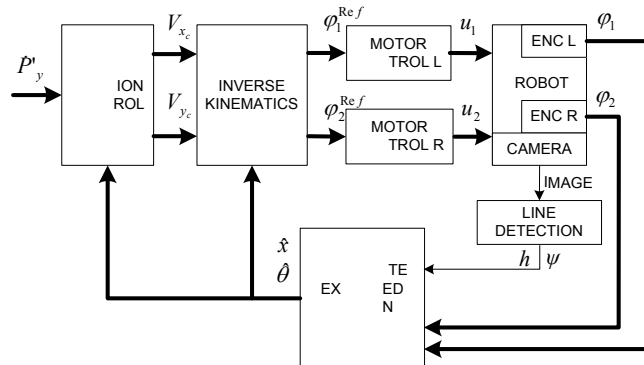


Figure 8: Control Architecture with MR-EKF

As can be seen in figure 9, the simulations give the same results as in the previous situation.

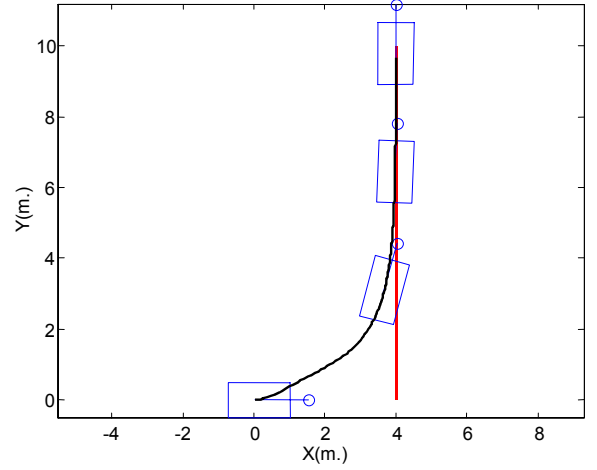


Figure 9: Line-Tracking with MR-EKF

Noise Robustness Analysis

A noise robustness analysis is performed to validate both multi-rate algorithms. Firstly, a white noise has been applied to the left wheel velocity measurement as depicted in figure 10, where it can be seen that in both cases, the noise is filtered. Moreover, in the case of a constant disturbance, the left wheel speed is sliding continuously and the resulting track (figure 11) for both algorithms demonstrates that line is followed with a constant lateral error, giving a bad estimation.

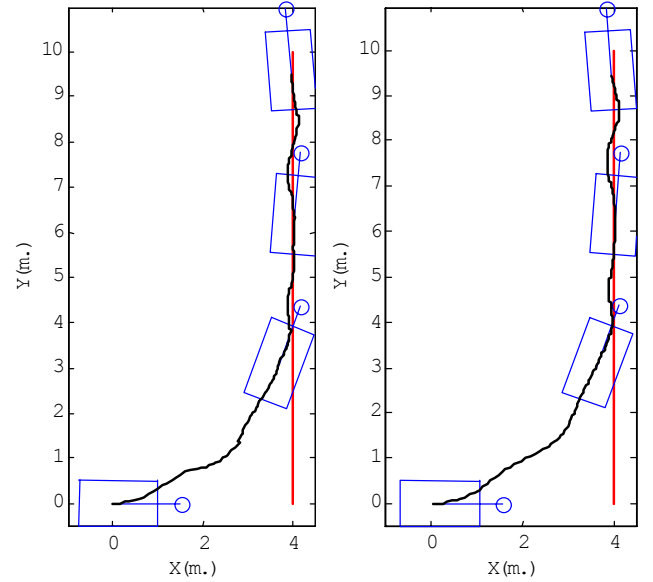
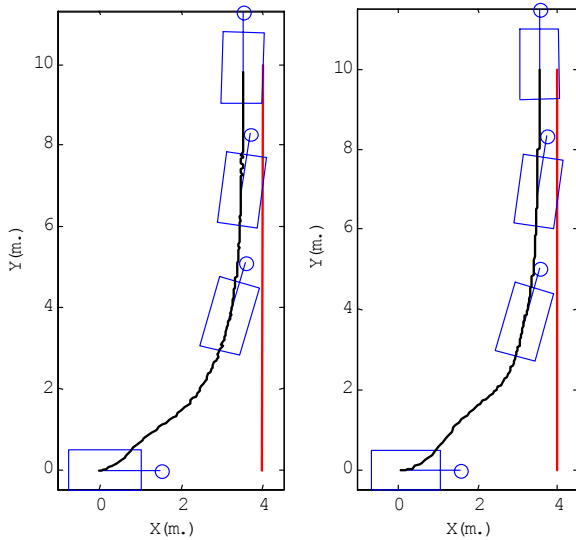
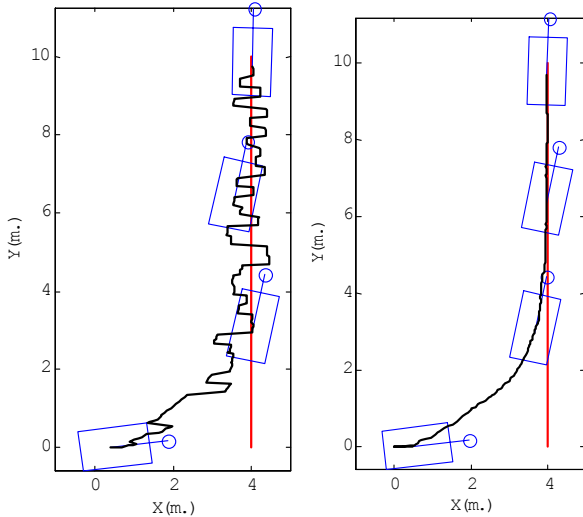


Figure 10: MR-Line Tracking with Input White Noises

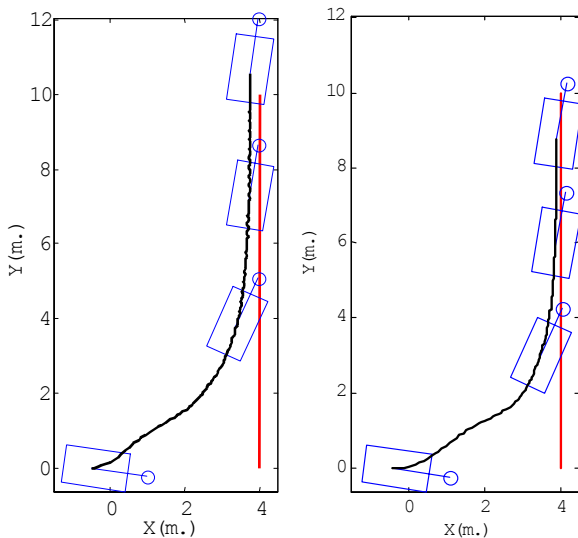
However, when a white noise is applied to both parameters h and ψ , the MR-Posture Estimation is strongly affected by noises on those parameters meanwhile the MR-EKF filters the noises. Figure 12 shows that the MR-Posture Estimation has a lateral error, while the MR-EKF follows the line without a lateral error. Similar conclusions can be extracted from Figure 13, where the line parameters are affected by a constant disturbance.



MR-POSTURE ESTIMATION **MR-EKF**
Figure 11: MR-Line Tracking with Constant Input Disturbance



MR-POSTURE ESTIMATION **MR-EKF**
Figure 12: MR-Line Tracking with Output White Noises



MR-POSTURE ESTIMATION **MR-EKF**
Figure 13: MR-Line Tracking with Constant Output Disturbance

CONCLUSIONS

In this paper, the problem of line-tracking for mobile robots using artificial vision have been presented based on a multi-rate structure. It can be observed that the robot behavior worsens when the image processing is forced at low single-rate sampling frequency. On the contrary, using a multi-rate structure, it is possible to combine the low frequency sampling rate associated to the image processing with the high frequency sampling rate of the internal sensors and control actions. That is, odometer sensor data on each wheel is combined with a camera processing data at different sampling rate giving multi-rate line-tracking algorithms.

Two different multi-rate control architectures have been implemented and tested: The first control architecture estimates the robot posture from the odometry (angular velocities of the wheels) using the camera data when available. The second control architecture uses a multi-rate Extended Kalman Filter (MR-EKF) to fusion sensor data (image and odometers). The results of both multi-rate algorithm are analyzed and compared. Both multi-rate algorithms give similar results except when the noise is applied to the measurements. In this case, the MR-EKF gives better performances because it considers the systems dynamic model.

Acknowledgments

This research has been partially funded by the Spanish Government (Ministerio de Ciencias y Tecnología) and FEDER. Research Projects DPI2000-0362-P4-05 and DPI2001-2689-C03-02.

REFERENCES

- Gracia, L. and J. Tornero. 2001. "Application of Kinematic Modeling Methods to Wheeled Mobile Robot", WSES Int. Conference on Mathematics and Computers in Physics.
- Gracia, L. and J. Tornero. 2002-1. "Vision-based System for Self-positioning and Calibration" *Advances in Systems Engineering Signal Processing and Communications*, 137-142.
- Gracia, L. and J. Tornero. 2002-2. "Kinematic Control System for Car-Like Vehicles" *Advances in Artificial Intelligence – Iberamia 2002*, 882-892.
- Marchant J. 1995. "Tracking of Row Structure in Three Crops using Image Analysis", *Computer and Electronic in Agriculture*.
- Sorenson, H., 1985. "Kalman filtering : theory and application", Ed.: *IEEE Corp.*, ISBN 0879421916.
- Tornero, J.; Salt, J. and Albertos, P. 1999. "LQ Optimal Control for Multirate Sampled Data Systems." 14th World Congress of IFAC, Peking (China).
- Tornero, J. 1985. "Non-conventional sampled-data systems modelling." *Control System Centre Report*, no 640/1985. University of Manchester (UMIST).
- Tornero, J.; Pizá, R.; Albertos, P. and Salt, J. 2001. "Multirate LQG Controller Applied to Self-Location and Path-Tracking in Mobile Robots." *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2001)*, Maui, Hawaii (USA).

REAL-TIME APPLICATION FOR CAR-LIKE VEHICLES BASED ON CAN BUS

Leopoldo Armesto, Santiago Fuster, Francisco Jiménez and Josep Tornero
Dpto. Ingeniería de Sistemas y Automática
Universidad Politécnica de Valencia
46022 Valencia (Spain)
Email: {leoaran,sanfusdi,frajilo,jtornero}@isa.upv.es

KEYWORDS

CAN bus, Kinematic Control, Potential Fields, Real-Time.

ABSTRACT

This paper describes a real-time application designed to control an electric vehicle with own-developed CAN nodes, distributed around the vehicle. A flexible data structure has been implemented to handle with CAN messages. The vehicle tracks a predefined trajectory in combination with an object avoidance algorithm. A simulator has been also developed for the validation of the data structure and the control algorithms. Both applications run under Linux.

INTRODUCTION

The electric car Bombardier NV, shown in Figure 1, is a fiberglass vehicle designed for leisure time. This vehicle is been used for applications such as object detection and collision avoidance, motion planning, control and data fusion.

A laser ranger, LMS200 from SICK, has been installed, at the front part of the vehicle. This sensor activates three digital outputs when an intrusion on three predefined fields occurs. Laser sensors also provide range information of a 180° scan with 0.5° and 1 cm precisions via RS232.

The vehicle is also equipped with a ring of 16 US and 12 IR sensors used as medium and short distance protection mechanisms, respectively. US sensors from Polaroid 600-Series are used to detect object presence up to 8m with a opening angular range of 30°. The IR sensors, GP2D02, transmit distance information up to 70cm. A GPS module is also installed in the car.

This vehicle has been automated with an on-board embedded computer connected to sensors and actuators, distributed around the vehicle, through CAN bus. For each sensor and actuator, it has been designed and implemented a specific CAN node. Each CAN node is based on a PIC μ Controller connected through the bus at 500Kbauds. The whole hardware architecture is described in Figure 2.

The steering wheel has been automated by introducing a commercial servo-assisted steering. Three CAN-nodes has

been implemented in order to control the steering wheel position: the DC motor driving the servo-assisted steering is controlled through a D/A CAN-node; the steering wheel position is measured with an absolute encoder CAN-node; and the torque applied to the power-assisted steering is measured with a A/D CAN-node.

For the speed control loop three CAN nodes have been designed: a Hall effect CAN-node which measures the vehicle speed, two A/D nodes for the accelerator and break measurement, as well as a D/A converter CAN-node for the electric motor actuation.

There are defined three possible working modes for each node: stop, running periodically and running on command request. The stop mode is used for the node configuration (default on start up), the periodical mode is used for normal operation, meanwhile the on command request mode is been used for debugging purposes.

Special attention has been paid to CAN-node priority structure. In this sense, digital signals provided by the laser are at high-priority. On the contrary, A/D accelerator pedal and to the A/D break pedal-nodes used just for monitoring are working at low-priority. Other nodes, such as those associated to IR and US sensors have the lowest priority due to the uncertainty provided. As it well-known, priority is established with the CAN identifier.



Figure 1: Electric Car

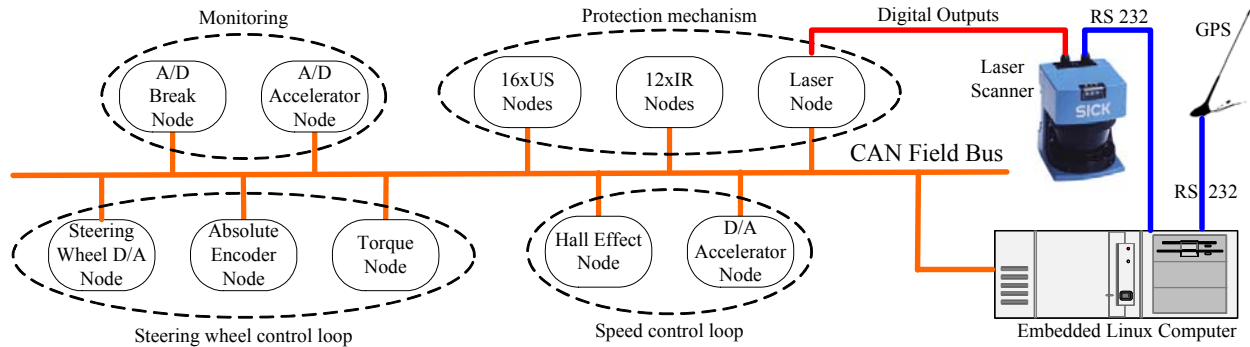


Figure 2: Hardware Architecture

CAN-BUS INTERFACE ARCHITECTURE

Controller Structure

A real-time application has been implemented to control the electric car. This application runs under Linux and communicates with CAN nodes, using CAN messages for sending and receiving data. A flexible information structure is required for handling and storing the data. In this sense, a blackboard data structure has been designed according to the sensors and actuators available in the vehicle. Nevertheless, this data structure is easily adaptable to other mobile robots controlled by CAN nodes. Handling with the blackboard, there are three tasks, as shown in Figure 3: the main process, the CAN reader and the CAN-writer. All source code has been written in ANSI-C for portability. This software architecture has been design for ease exchange to different robot applications.

In particular, the blackboard is composed by three lists: The Node List (NL), containing a short description of the developed nodes; The Incoming Frame List (IFL), storing frames received from each node; and The Outcoming Frame List (OFL), which keeps frames that have been already sent to the CAN bus. It is interesting to remark, that for each node there are two pointers in the NL indicating the last frame sent in the OFL or received in the IFL. These two boards, OFL and IFL, are sets of circular buffers containing the last hundred frames of each node.

The main process is launched when the application is executed by the user. It launches the other processes: CAN reader and writer. It also allocates and releases the shared memory, installs and uninstalls the CAN driver. Initially, the main process creates the blackboard structure in a shared memory and initialize the pointers to the beginning of the circular buffers. It also programs the nodes in periodic mode associating their corresponding periods and offsets. At the beginning, the main process synchronizes the CAN-nodes sending the synchronizing frame just after they have been configured. Then, it enters in a periodic loop, where reads sensor data from the IFL, applies the corresponding control algorithms and sends back the references to the nodes. Then the process is suspended until next period.

The CAN-writer process sends messages to the CAN bus through the driver. This process is suspended until the main process requires to send a message. It also writes the messages on the OFL, updating the corresponding pointer in the NL.

Similarly, the CAN-reader process is also always suspended, waiting for an incoming frame from the CAN bus driver. It writes data on the IFL board updating the pointer. This process can also detect if a node has fallen notifying the event to the controller.

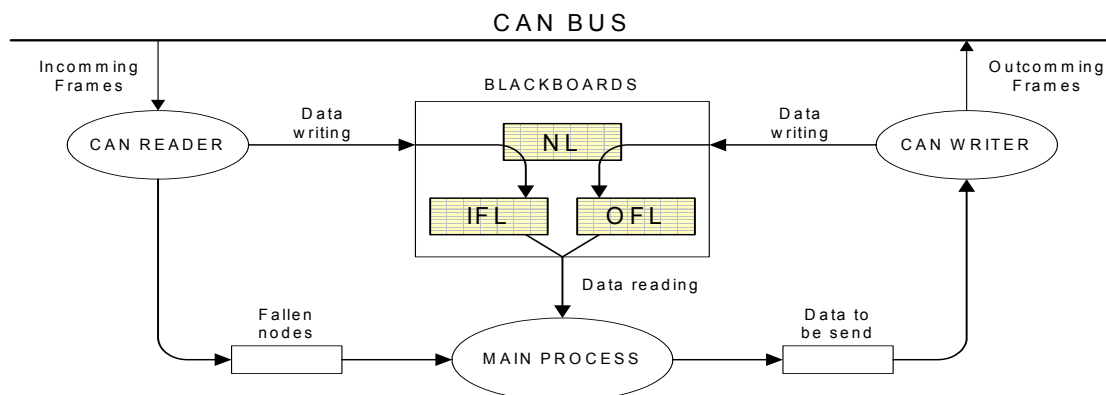


Figure 3: CAN-Bus Interface Architecture

Vehicle Simulation

In order to validate the different control algorithms and the software architecture described above, a simulation application has been implemented based on two computers: vehicle simulator and controller. In one computer, the simulator, the vehicle direct kinematics, the sensor and actuator models together with the CAN-node structure have been simulated. On the other computer, the controller, carries out its task as if it were connected to the real vehicle.

For the simulation of the electrical car shown in Figure 1, the Hall effect sensor for the wheel speed, the absolute encoder for the steering wheel and the sonar rangefinders have been modeled.

It is interesting to remark that the software control architecture shown in Figure 3 has been implemented on two computers just as a way of validating the CAN-Bus interface.

KINEMATIC CONTROL

At this point, a kinematic control has been implemented instead of a dynamic control, in order to avoid: complexity, sensitive to uncertainties in model parameters, etc.

The electric car is based on the well-known Ackerman's mechanism, being its kinematic model equivalent to the tricycle one, depicted in Figure 4. This model is composed by three wheels, being one of them the steering wheel. The kinematic control calculates the wheel velocities (in the rotation and orientation axes). The vehicle state is formed with $\rho' = [x' \ y' \ \theta' \ \beta']^T$ and the control input $u = [v \ \beta]^T$.

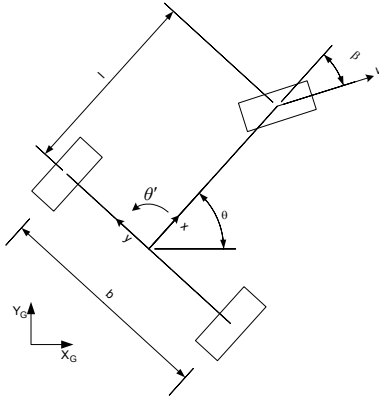


Figure 4: Tricycle Kinematic Vehicle

The direct kinematic model relates the speed measurements (v) from Hall effect sensor and the wheel position (β) from the absolute encoder, and it is computed as follows (Ollero, 2000):

$$\rho' = B(\rho) \cdot u$$

$$\begin{bmatrix} x' \\ y' \\ \theta' \\ \beta' \end{bmatrix} = \begin{bmatrix} \cos \theta \cdot \cos \beta & 0 \\ \sin \theta \cdot \cos \beta & 0 \\ -\frac{\sin \beta}{l} & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v \\ \beta' \end{bmatrix}$$

Where the vehicle's posture can be easily computed integrating the state vector:

$$\rho = \int \rho' \cdot dt$$

For the inverse kinematic model, we have used the pseudo inverse of $B(\rho)$, as follows:

$$u = (B^T(\rho) \cdot B(\rho))^{-1} \cdot B^T(\rho) \cdot \rho'$$

Nevertheless, it presents numerical computation problems and therefore we prefer to use a simplified inverse kinematic model.

$$\begin{bmatrix} v_r \\ \beta_r \end{bmatrix} = \begin{bmatrix} \sqrt{x_r'^2 + y_r'^2} \\ \theta_r - \arctan\left(\frac{y_r'}{x_r'}\right) \end{bmatrix}$$

Note we use sub-index r in variables indicating that is a reference value.

For the kinematic control, different well-known techniques have been considered, such as those based on geometric methods (Ollero, *et. al.*, 2000), linear approximations based on non-linear techniques (guaranteeing the system stability through the existence of a Lyapunov function) (Lyshevski, Nazarov, 2000) or state feedback linearization with classical reference tracking. In particular, a PD tracking point controller has been designed as follows:

$$\begin{bmatrix} x'_r \\ y'_r \end{bmatrix} = \begin{bmatrix} x'_{ref} \\ y'_{ref} \end{bmatrix} + \begin{bmatrix} a_x & 0 \\ 0 & a_y \end{bmatrix} \cdot \begin{bmatrix} x_{ref} - x \\ y_{ref} - y \end{bmatrix} \quad (1)$$

where the tracking point is given by $[x_{ref}, y_{ref}, x'_{ref}, y'_{ref}]^T$ and parameters a_x and a_y determine the convergence to the tracking point.

The control architecture implemented for the kinematic control is depicted in Figure 5.

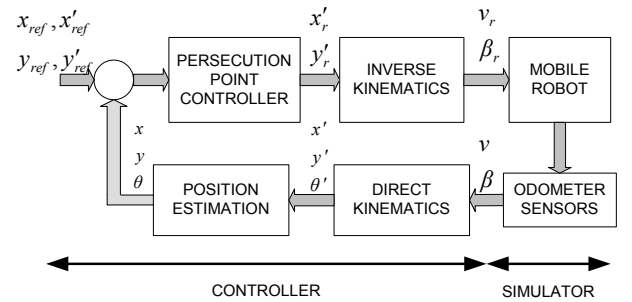


Figure 5: Control Architecture

In simulation, a reference track application has been implemented. The velocity reference and steering wheel position are sent from the controller to the simulator and on the other hand, the simulator returns the vehicle velocity and the wheel position. The resulting path track for a given reference is depicted in Figure 6, where there have been included two obstacles, for latter implementation of object detection and collision avoidance algorithms.

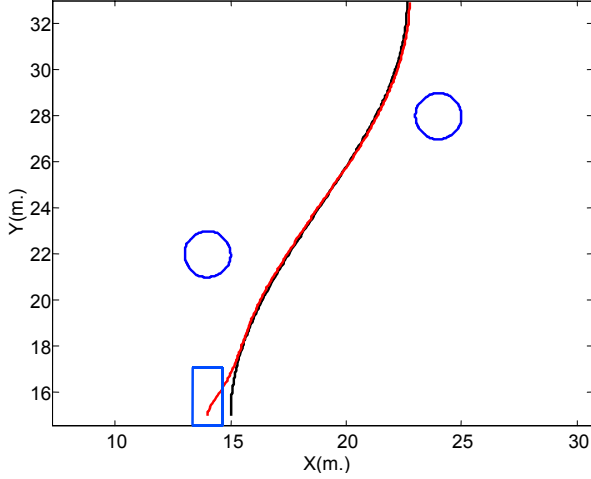


Figure 6: Path Tracking without Range Sensors

SONAR RANGERS SIMULATION

In order to perform obstacle avoidance, it is necessary to obtain environment information. This is performed with the sonar sensor measurements, obtained by simulation based on the robot position and its environment. It is calculated for each sensor the possible detections of all objects in the simulator application.

In this sense, for each sensor, we calculate its position and orientation $(x_{gs_i}, y_{gs_i}, \alpha_{gs_i})$ with respect to the global coordinates system, by using homogeneous matrices,

$$(x_{gs_i}, y_{gs_i}, \alpha_{gs_i}) = (R(\theta) \cdot T(x, y)) \cdot (R(\alpha_{ls_i}) \cdot T(x_{ls_i}, y_{ls_i}))$$

where (x, y, θ) is the position of the vehicle with respect to the global coordinate system and $(x_{ls_i}, y_{ls_i}, \alpha_{ls_i})$ is the position of each sensor with respect to the local robot coordinate system.

We suppose that the surfaces to be detected are specular, and therefore a sonar beam is reflected with the same incident angle with respect to surface normal vector. If the angle formed between the surface $\alpha_{v_{n,j}}$ and the sensor α_{gs_i} differs in less than half of the beam angle (φ_i) , then the ultrasonic wave returns to the sensor and it can be detected.

$$\alpha_{dif_{ij}} = \alpha_{gs_i} - \alpha_{v_{n,j}} - \pi$$

$$|\alpha_{dif_{ij}}| \leq \frac{\varphi_i}{2} \Rightarrow \text{Object detected}$$

To simulate the environment, we have defined two kind of obstacles: flat surfaces, such as walls, and circles.

Flat surface detection

A flat surface is defined by its starting and ending points, and its unit normal vector \vec{v}_{n_j} . The intersection point (x_{ij}, y_{ij}) between the flat surface and the line perpendicular to the sensor is geometrically calculated as see in Figure 7. Then, the measured distance is computed as follows:

$$d_{s_{ij}} = \sqrt{(x_{ij} - x_{gs_i})^2 + (y_{ij} - y_{gs_i})^2} \cdot \cos(\alpha_{dif_{ij}})$$

$$d_{s_i} = \min_{ij \rightarrow \text{valid}} (d_{s_{ij}})$$

Those points which do not correspond within the definition limits of the surfaces are not considered.

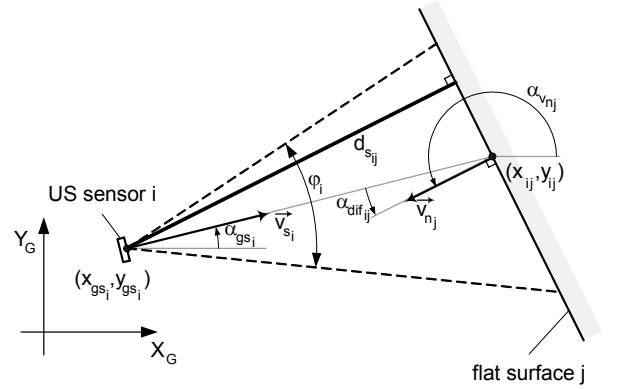


Figure 7: Flat Surface Detection

Circular surface detection

If the object detected is circular as see in Figure 8, the distance between sensor and circle is determined as follows:

$$d_{s_{ij}} = \sqrt{(x_{c_j} - x_{gs_i})^2 + (y_{c_j} - y_{gs_i})^2} - r_j$$

$$d_{s_i} = \min_{ij \rightarrow \text{valid}} (d_{s_{ij}})$$

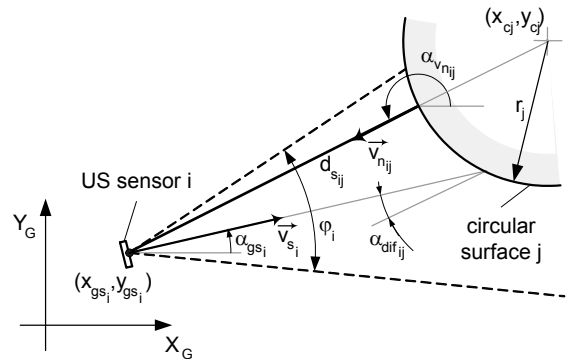


Figure 8: Circular Surface Detection

OBJECT DETECTION AND AVOIDANCE

Mobile robots require somehow an object detection and collision avoidance algorithm in order to be able to navigate amidst an unknown environment with moving objects. We have focused the attention to reactive algorithms, which provide faster response to the perception of the environment. It is possible to find many contributions dealing with reactive methods (Borenstein and Koren, 1989), (Brooks, 1986) and (Kahtib, 1986, 1987).

In the application described in this paper, potential fields (Kahtib, 1986, 1987) are used to validate the data structure used for both the controller and the simulator. Based on a predefined environment, the simulator calculates the sonar sensor measurements as described in the previous section. Then, data is transferred to the controller which computes the potential fields algorithm according to the measured distances. Then, a repulsion vector \vec{r}_{rep}^r is computed as follows,

$$\vec{r}_{rep}^r = \sum_{i=0}^{n-1} \frac{k_i}{\|\vec{d}_{s_i}\|^3} \cdot \vec{d}_{s_i}$$

The repulsion force is then referred to the global system coordinate, using an homogeneous transformation matrix, and converted into a velocity repulsion vector for the control loop.

$$\begin{pmatrix} x'_{rep} \\ y'_{rep} \end{pmatrix} = \vec{r}_{rep} = R(\theta) \cdot \vec{r}_{rep}^r$$

This repulsion force has been integrated in the persequition point controller (1), giving:

$$\begin{bmatrix} x'_r \\ y'_r \end{bmatrix} = \begin{bmatrix} x'_{ref} \\ y'_{ref} \end{bmatrix} + \begin{bmatrix} a_x & 0 \\ 0 & a_y \end{bmatrix} \cdot \begin{bmatrix} x_{ref} - x \\ y_{ref} - y \end{bmatrix} + \begin{bmatrix} x'_{rep} \\ y'_{rep} \end{bmatrix}$$

with the control architecture,

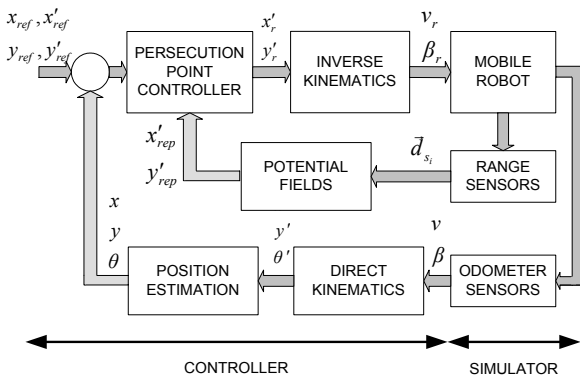


Figure 9: Control Architecture with Range Sensors

Figure 10, shows the same path tracking reference as shown in Figure 6, with the potential fields algorithm implemented. Figure 10 denotes that both objects have been detected by

the sensors and in consequence the mobile robot modifies its path.

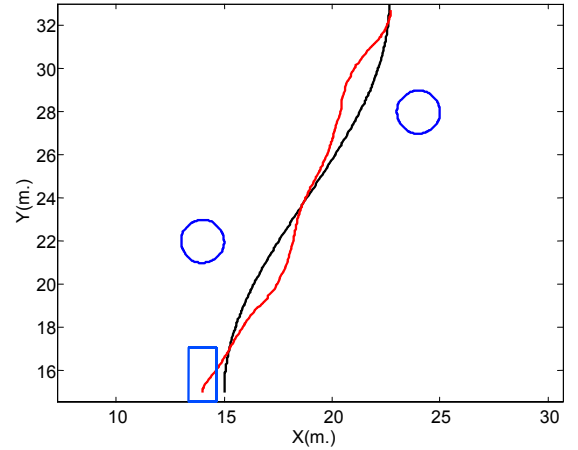


Figure 10: Path Tracking Using Range Sensors

CONCLUSIONS

We have implemented a flexible software and hardware architecture, for validating mobile robot control algorithms running in real-time. A distributed sensor and actuator architecture based on CAN-nodes has been physically constructed. A simulator has been also developed for the validation of the data structure and algorithms. Further research work is oriented in the real implementation on real vehicle.

Acknowledgements

This research has been partially funded by the Spanish Government (Ministerio de Ciencia y Tecnología) and FEDER. Research Projects DPI2000-0362-P4-05 and DPI2001-2689-C03-02.

REFERENCES

- Armesto, L., Tornero, J., Torres, J.C., 2003 "Hardware Architectures For Mobile Robots", Accepted on Symposium on Robotic Control (SYROCO'03).
- Ollero, A., Robótica, 2000, "Manipuladores y Robots Móviles". Marcombo Boixareu (Ed.) ISBN: 84-267-1313-0, 2000.
- Lyshevski, S.E., Nazarov, A., 2000 "Lateral Maneuvering of Ground Vehicles: Modeling and Control, *Proceedings of the American Control Conference*.
- Borenstein J., Koren, Y., 1989, "Real-time obstacle avoidance for fast mobile robots". *IEEE Transaction on Systems Man and Cybernetics*, Vol. 19, pp. 1179-1187.
- Brooks R. A., 1986, "A Robust layered control system for a mobile robot". *IEEE Journal of Robotics and Automation*, RA-2, pp. 14-23.
- Khatib O., 1986, "Real time obstacle avoidance for manipulators and mobile robots". *IEEE Journal of Robotics and Automation*, Vol. 2, pp. 90-98.

Industrial Robotics

REAL-TIME SIMULATION OF AN INSPECTION ROBOT WITH A COMMERCIAL PHYSICS ENGINE

Jan Andries Neuhoefter
Fraunhofer Institute for Autonomous Intelligent Systems (AIS)
Schloss Birlinghoven
53754 Sankt Augustin,
Germany
E-mail: Jan.Neuhoefter@ais.fraunhofer.de

ABSTRACT

A 3D real-time simulation of a sewer inspection robot called MAKRO is described. In order to achieve an optimal combination of speed, accuracy and stability, the commercial physics engine Havok™ by Telekinesys™ in its newest version 2 has been employed. It handles both real-time mesh collisions and physical characteristics, originally coming from the 3D gaming industry. In contrast to conventional tools, Havok™ does not have a common graphical user interface in which you compose your simulation, but is an SDK (System Development Kit) in C++. This demands development on a more abstract level, but in the same way offers extended options. MAKRO's actuators as well as all relevant sensor systems have been modeled, always in consideration of the robot's physical attributes. First results encourage to use this simulation method for other robots and automotive systems as well.

THE ROBOT

MAKRO is an articulated inspection robot of worm-like shape and designed for pipes of 30 to 60 cm of width (Klaassen 2002). It is 2 m long and consists of six segments connected with 5 active joints. Each joint is a combination of 3 rotating discs. The two outer discs are in parallel to their adjacent body parts whereas the disc in the middle has its axis in a 45° angle to each of its neighbors. So in principle each joint has the flexibility of a ball joint, but the modeling is more sophisticated. All in all we get 21 degrees of freedom and a weight of about 50 kilograms. Worth to mention is that MAKRO runs completely autonomously which means that it is not tied to cable that connects it to a surveillance unit but unleashed and carries all the resources it needs on board. Navigation and motion control is done on a PC/104 CPU with Linux-RT as operating system. Wireless LAN is attached to the system so it can communicate with any other computer within range.

Besides the internal sensors like pulse counters for odometry and optical sensors for reading joint angle encodings, MAKRO makes use of 5 infrared distance sensors at each end (head and tail are identical), pointing up, down, left, right and straight ahead. A laser system projects a pattern onto the pipe walls which, in combination with the stereo camera pair, allows attitude detection in relation to the pipe axis (Kolesnik 1999). Perpendicular sensors tell about the absolute attitude.



Figure 1: Sewage Inspection Robot MAKRO

Why a simulation?

Due to its dimensions and weight, the robot is not easy to handle and operate in a laboratory (although there is an indoor-facility to perform tests). Its destination environments are pipes of diverse bores with obstacles and stairs within. So a testing scenario in reality is always set up with an elaborate effort only. Anyway, navigation routines have to get improved constantly. Furthermore, there are problems and questions as a result of the robot's bodywork when being confronted with displaced and broken pipes. In consequence, the simulation saves money, material and time. Furthermore, it is useful for technology demonstration purposes.

THE SIMULATION

The simulation is a virtual representation of the real robot with regards on the robot's physics and, as far as possible, functional devices. So, not only geometries, mass and friction have to be modeled but also actuators (motors and drives) and sensor systems (distance sensors, attitude sensors etc.) have to be implemented as far as they concern the navigational aspects. A very important feature of the simulation from the beginning on was to make it easily and quickly modifiable, not only concerning the robot itself but also the environment it moves through. And this in a way that not only 3D programmers with special knowledge are capable of accommodation, but - like in this case - also

control developers who rather like to focus on control algorithms and not really on mesh collision issues, for example. Originally, the simulation of the robot has been done with the simulations tool ADAMS™ by MSC Software™ (Klaassen 2000). It offers a high level of accuracy and excellent analysis capabilities, but momentarily does not meet the project's requirements concerning speed for coming use cases.

Havok™ – Realtime 3D Physics Engine

Havok™ is an object-oriented framework, or to be more precise, an SDK in C++. As a matter of principle, the user creates a three-dimensional world, adds objects to it, assigns physical properties to these objects and then lets the world step forward in time. When talking about real-time simulation, we think of continuous simulation and display. In practice this means performing single simulation and display operations several times per second. While doing so, Havok™ bears in mind the specified geometries of the objects, physical attributes, constraints, occurring forces and user interactions. Very important in this context is collision detection which has to come about quickly as well as with adequate precision, and that without neglecting stability. Construction of all geometries can be done manually by defining each object vertice by vertice. Alternatively, Havok™ offers an add-on for the well-know 3D designing tools 3D Studio Max™ by discreet™ and Maya™ by Alias/Wavefront™. These add-ons, called Exporters, allow to export geometries, constraints and motors created within the design environments of these tools. A preview function helps to determine the correctness of all construction beforehand, subsequently, the objects can be exported into a file which then is easily importable to the simulation itself.

Havok™ provides cross-platform display libraries which are sufficient for prototyping and simple demos. You can define the screen resolution as well as lights and camera settings. It is also possible to apply materials with color, alpha value and textures. Nevertheless, it is recommended to deploy a third-party renderer for optimal visual results. The screenshot shown in figure 5 is based on the standard display libraries.

Architecture of the simulation

Generally, the simulation's design is based on a server-client style where the simulation itself is the server and the control device is the client (see figure 2). This concept is reasonable since the robot control unit always *requests* the real actuators to run while motion of the robot is an *event* as well as sensor signals are *events*. The idea behind this is to modularize the simulation in a way that it completely replaces the real robot as a whole.

Since Havok™ is available for Windows as well as Linux, the choice of the Operating System is a question of user-specific infrastructure. This fact makes it easier to embed the simulation into an existing development framework. In this case, the Windows-version of Havok™ has been chosen firstly, a compilation for Linux will probably

follow. The control team which primarily uses the simulation has combined it with "MCA2 – Modular Controller Architecture 2" (Scholl 2000), a modular, network transparent and real-time capable C/C++ framework for controlling robots and other kind of hardware. This framework offers support for Linux, Windows and Solaris. Nevertheless it is also possible to integrate the simulation into a completely self-made framework.

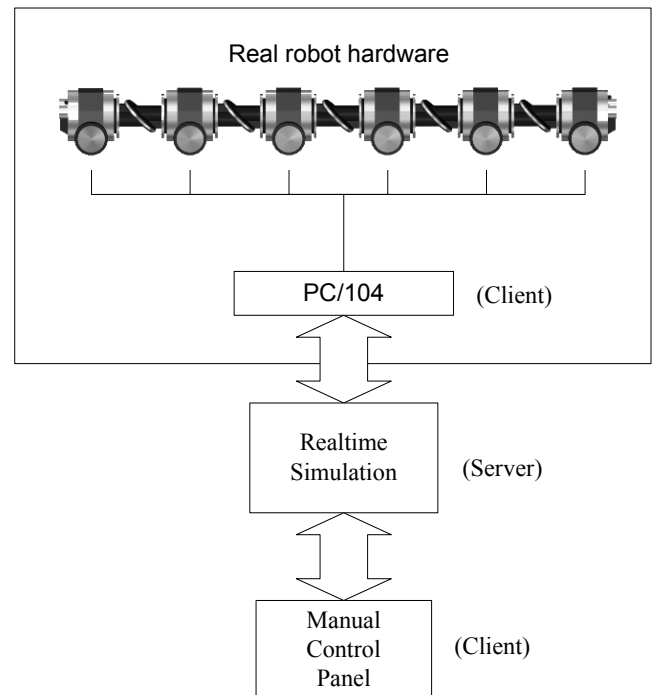


Figure 2: The simulation's architecture

The client could be the actual PC/104 on the back of the robot (including control and navigation algorithms), but it could also be a dialog panel with manual access. This strongly depends on the complexity and the degrees of freedom of the simulated system. Since MAKRO has 21 degrees of freedom, manual steering (directing each single motor) is hardly possible, except for testing purposes. Nevertheless it can facilitate fine-tuning and debugging enormously and is an essential part of the simulation, particularly in the early phase of its creation. And last not least, its functionality and design can serve as some kind of template for the following access modules.

Figure 3 depicts a screenshot of a manual control panel programmed under Linux to test the simulation and to seize the actual communicational situation (MAKRO's control routines run with RT Linux as Operating System). In contrast to the control team that uses MCA2, in this case an integrated self-made communication architecture (with sockets) has been created, just to show the flexibility of the simulation. The graphical user interface has been programmed with Trolltech's Qt™ (Dalheimer 2002), a multiplatform, C++ (continuously object-orientated) application framework with versions for Linux/Unix, Windows and Mac OS X and so is easily portable onto other platforms (for Linux, Qt is provided under the GNU General Public License which allows to use it for free as

long as it is open source; a fee has to be paid only for commercial distributions).

Basically, this client directly connects to the simulation server via TCP/IP and displays the current angles and drive speed settings as well as sensor values. Requests can be sent for particular motors and sensors as well as for all at once. The communication protocol is kept simple and compact to ensure a minimal network charge.

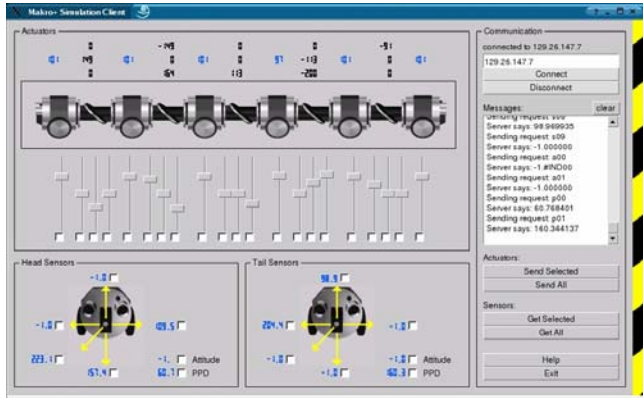


Figure 3: The manual control panel for Linux

Server and client need not necessarily run on different machines, but by a separation of those two, the server can utilize the machine's whole CPU power while data analysis and navigational calculations may proceed on a different computer. Ideally, the same computer which directs the real robot parts (in this case the PC/104 on the back of the robot) is just connected to the simulated (virtual) robot. This makes testing and development of new algorithms quick, easy and save.

Features of the simulation

The simulated world holds some standard options which can be modified due to a performance optimization. You can, for example, set the broadphase size (important for collision considerations), the collision tolerance and the solver type. The solver is responsible for calculating the necessary changes to reduce error for all constraints in the simulation, each solver type provides a different speed/stiffness balance.

The simulation's performance strongly depends on the complexity of the objects it has to deal with. The more vertices and therefore polygons they consist of, the more calculations have to be done. So a reasonable accuracy-speed-ratio has to be found to get best results. It is obvious that the more powerful the deployed computer is, the less problems will occur with complex calculations. You do not need to graphically output the simulation in any case, but if you want, high performance 3D acceleration is recommended.

As mentioned above, there are basically two ways to create geometries for a Havok™ simulation. The first one is more demanding at the beginning, but lets you influence your whole setup quicker afterwards, namely explicitly defining everything in C++ code. This means to create the objects involved into the simulation vertice by vertice which can

get exertive, especially when dealing with non-standard geometries. This method gives you full overview and control, and, depending on your coding skills, greatly facilitates modifications. Strictly object-orientated development helps to keep track on the whole assembly.

The second method is using a design tool with export function which is of more visual nature and especially recommended when designer and coder work separately. Following this path, you have the full designing capabilities of those professional design tools for your geometry creation. In addition, the plug-in offers special functionality which later on takes effect in the simulation like constraints, motors, masses and friction. Figure 4 shows a screenshot of a simple pipe environment for the MAKRO simulation in discreet's "3D Studio Max"™ (see figure 5 for an impression of how this environment looks like in the simulation itself, rendered with the standard output device which comes with Havok™). On the left side, you see the additional Havok™ exporter toolbar. The modus operandi for using this tool is simple: just create geometries with more or less advanced modeling techniques (here again: the less complex, the faster simulated), assign constraints if necessary, attach active elements like motors, check the whole setup with the preview function of the exporter module and finally export everything to a file. The file format can be ASCII (large, human-readable, platform-independent), binary (reduced file size, non-human-readable, machine-dependent to PC) and Xtra (for use in combination with Macromedia's Director™ which is not subject of this paper). The following and final step is to import the contents of the file into your C++ programming environment by a few lines of code. Subsequently, you are able to access all objects by their names you gave them in 3D Studio Max. Of course, you can import objects and environments on runtime to change the scene whenever you like.

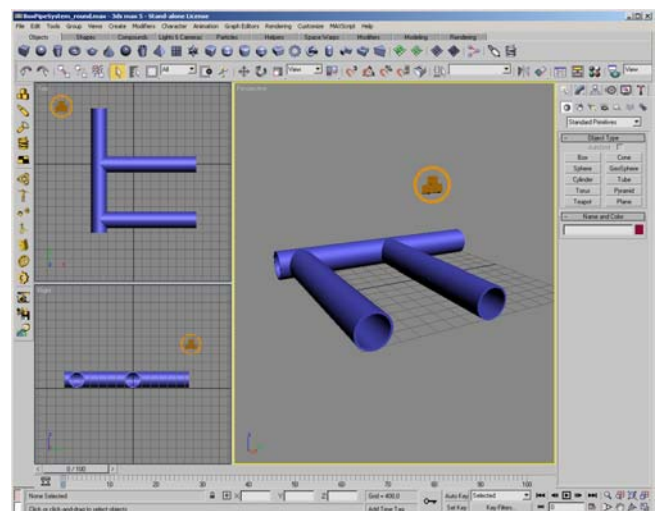


Figure 4: Environment design with 3D Studio Max™

For the MAKRO-simulation, a combination of both methods is most convenient. The robot is purely coded while the environment he moves through is designed with a tool. By this way, the actual applicant of the simulation (who is not necessarily the originator) can easily create own

worlds without knowing a bit about Havok™ or even programming. MAKRO moves through all kinds of pipes with varying width and material which has enormous effect on its freedom of movement. Sudden confrontations with handicaps and humps are common for the real robot (displaced and broken pipes, blockages of all kinds). Setting all this up would be hard when done without a graphical user interface, so 3D Studio Max™ and Maya™ come right in place here. Basic modeling techniques in 3D Studio Max™ are to be learned in short time.

As an effect of compact programming, the virtual robot is completely customized and parameterized within seconds. The number of segments can be altered as well as the properties of each single part, and not only mass and dimensions, but also friction values. This can enormously influence future design considerations because now you see beforehand whether canting of the robot with a certain segment length in a pipe is more probable than with another length, for example. What about another wheel radius, thickness or material to effectively antagonize slippery and defective surfaces? Does it help to enlarge the clearance between the wheels? Would it be better to relocate the position of the rechargeable batteries to influence the center of gravity in future versions of the robot? All these considerations and much more can directly be taken into account.

Havok™ offers a variety of different joint types and motors. Especially made for automotive fields of use, the Havok™ Vehicle Kit which is a subset of the SDK, helps to develop physically-based vehicles which fully integrate with the engine. In order to simulate the robot's actuators, the drives for the wheels and the more special 3-disc-joints had to be created. In case of the latter, the concept of a powered hinge joint is adapted to get a behavior close to the real exemplar. This kind of joint has no physical representation itself like a door hinge, for example, but rather describes a mutual constraint of two geometries. Since the torque which is applied to that hinge (to make it a POWERED hinge joint) directly affects the angle between the two geometries involved, we get an action-reaction-behavior which is important for the virtual robot's capabilities to bend and to level.

A restriction to the real robot is the fact that it is not allowed to twist too many times, because the cables inside the joints would be torn after a few rounds. To make allowance for this phenomenon, the hinge joints can be limited to a certain angle-range. Generally, Havok™ even allows constraints to be defined as to be breakable which includes a break off in case of overstressing.

Regarding sensor systems, devices for odometry and reading joint angle encodings is not difficult to represent in a 3-dimensional world since all the transformational information (translation and rotation) of each object can easily be obtained and processed. Thus, a sensor which gives information about the perpendicular direction is not really hard to implement. The distance sensors are simulated in a way that ray-scans are done in the 3D-world which means to define two points within space which

determine a unique straight line in space. If the line crosses a geometry, the scan is positive, else it is negative.

Besides the ten distance sensors (five at each end), the robot uses a laser system (described in paragraph "THE ROBOT" at the beginning of this document) to assert its attitude in relation to the pipe axis. The according simulated sensor system gives the same value, but works differently, namely more in way of some analytical geometry calculations and so avoids the quite elaborative process of video image analysis (what the real system does). To make the sensors more realistic, one way would be just to provide them with a disturbance factor. This could be a time-dependent Gaussian variable, for example.

In figure 5 we have a screenshot of the actual simulation. As mentioned above, it currently runs under Microsoft Windows. In the lower left corner, you can see a simple self-made control panel which allows to open the port for network clients to connect. Furthermore, you can set most important values like gravity and delta timestep. Last not least it gives control of the externally loaded environment file created with 3D Studio Max™ beforehand. Extending this panel with more functionality would be a piece of cake, but it is intentionally not overloaded for leanness reasons. Looking at the 3D scene itself, the robot within a pipe is depicted. The green (1) and red (2) lines on the head and the tail of the robot represent the distance sensors. Green means "no object within range" while red indicates an object detection. This could be turned off, of course, but is useful for functionality monitoring. The blue (3) line indicates the pipe axis of that pipe segment the sensor is currently located in. The grid is used for orientation (like the artificial horizon in airplanes), because especially when dealing with pipes with inclination, the user may lose orientation for where ground zero is. As a nice side effect you get a scalable measure grid (here one unit is 0.5 meters).

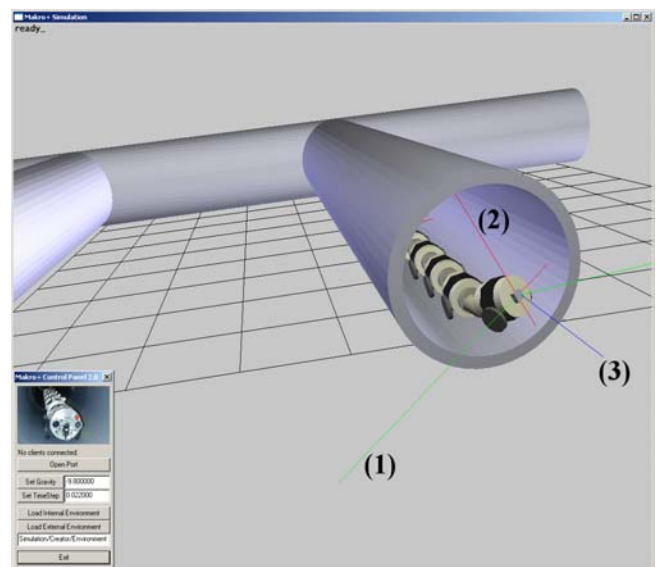


Figure 5: Screenshot of the simulation

Another interesting feature is the click-and-drag option. By clicking and dragging an arbitrary object of the scene, a

force relative to the dragged distance is inflicted on the object, of course with simultaneous consideration of its motion constraints. This is very helpful for artificial disturbance generation since you are verbatim able to manually encroach upon the scene.

Of course you can freely rotate, pan and zoom the scene to get any perspective you like. With the standard output library, it is also no problem to make objects (partially) transparent (an advantage you can hardly provide with real test-pipe setups). This allows setting up closed complex pipe systems without losing overview in order to approach real pipe system configurations as closely as possible.

SUMMARY

A 3D real-time simulation for a real multi-joint inspection robot called MAKRO was described in this paper. The simulation is implemented under C++ with the help of Havok™, a commercial 3D physics engine by Telekinesys™ which allows specification of physical attributes for the robot (directly programmed as a concept of collaborative classes) as well as the environment (designed with 3D Studio Max™ and later on imported to the simulation system). Always bearing in mind speed, stability and accuracy, the system's performance depends on the complexity of the simulated scene (robot and environment) and can directly be influenced by world parameters. Actuators as well as several kinds of sensor systems are simulated in real-time. The design of this simulation is of server-client-style, the implementation works with network support and is currently cross-platform (Windows and Linux). A tentative version of the simulation is already in use by the MAKRO robot control team and first results are very promising.

REFERENCES

- Klaassen, B.; H. Streich; E. Rome; F. Kirchner. 2002. "Simulation and Control of the Segmented Inspection Robot MAKRO". In *VDI Berichte NR. 1679, 2002*. pp. 611-616.
- Klaassen, B.; F. Kirchner; D. Spennberg. 2000. "Integration of robot control programs into ADAMS™ including sensor feedback". In *14th European Simulation Multiconference 2000*. pp. 259-262.
- Kolesnik, M. 1999. "View-based Method for Relative Orientation in the Pipe". In *Proceedings of SPIE Sensor Fusion: Architectures, Algorithms and Applications III*, SPIE – The intern. Society for Optical Engineering USA 1999, pp. 70-78
- Scholl, K.-U.; V. Kepplin; J. Albiez; R. Dillmann. 2000. "Developing Robot Prototypes with an Extendable Modular Controller Architecture". In *Proceedings of the international Conference on Intelligent Autonomous Systems, Venice 2000*, pp. 67-74
- Dalheimer, M. K. 2002. "Programming with Qt – Writing portable GUI applications on Unix and Win32". O'Reilly publishing, 2003. ISBN: 0-596-00064-2

AUTHOR BIOGRAPHY

JAN ANDRIES NEUHOEFER was born in 1977 in Unna, Germany. After achieving his general qualification for university entrance in 1996, he did his military service and subsequently went to the University of Stuttgart where he is an enrolled student of Technical Cybernetics since 1997. Momentarily, he is doing his diploma thesis at the Fraunhofer Institute for Autonomous Intelligent Systems in Sankt Augustin, Germany, and will presumably obtain his degree as a graduate engineer in the middle of 2003.

Ruling Robots by the Activity Patterns of Hierarchical SOMs

Matthias Reuter
CUTEC Institute
Leibnizstr. 21-23
D-38678 Clausthal-Zellerfeld
E-mail: reuter@informatik.tu-clausthal.de

KEYWORDS

Robots, SOM, Computing with activities

ABSTRACT

In this paper a first example is given, in which way robots can be steered by global activity pattern of self organizing maps (SOMs). In the context of the "Computing with Activities" a hierarchical neural network structure was trained which codes the diverse situations of (simulated) soccer playing robots not by a winner neuron but by the whole activity structure of the SOMs.

THEORETICAL FRAMEWORK

Even if a large number of neural nets is known none of these nets can solve some basic features which biological neural nets can do. Above all none of the simulated nets can join similar classification or identification concepts to new 'meta-concepts (chunks)'. This process, called 'chunking', is well known from neurology and even very primitive clusters of biological neurons can do it. It is a favourite principle of these chunks that on the one hand the different parts of these combined concepts can be activated individually by a special input-pattern but on the other hand also the whole chunk can be activated if only single parts of the input pattern activating chunked information is presented to the net. Another problem of the common neural nets is that classification results are solely represented by the activity of the neurons of a special output-layer, why even the power of classification of the whole net is restricted by the number of the present 'grandmother'-neurons of the output layer. That's the reason why it seems that we are working with a bottle neck which strips the classification behavior of the neural nets to a very simple and localized representative. May be for that reason nature does not use such kind of neurons in brain but the activity patterns of an association of neurons which are active at the same time to encode classification behaviour and classification results. By this way it is possible:

- to integrate data from different senses in one global activity pattern,
- to integrate more and more information (aspects) in one activity pattern by modifying its structure step by step,
- to chunk diverse classification concepts by combining so far different activity patterns.

Using the activities of all neurons of a simulated neural net for analyzing its classification behavior means to be aware of the fact that different kinds of activity patterns occur in a net. At first the activities of the inter-neural weights or the activities of the neurons only or both together can be supervised. For selecting the best supervising strategy it may help to take a view on, how information is coded and activated in biological nets.

Biological nets (wet nets) encode information in two kinds [1]:

- The structure of the activity of a neural network at juncture k encodes the classification result/classification concept of the momentary constellation of the senses data.
- The firing rate of the neurons encode the intensity of the sense data.

To fulfill the first requirement, it seems to be obvious to use the activities of the neurons of a self-organizing map (SOMs), as SOMs form symmetric a 2-dimension grid. On such a grid activity pattern can be analysed and visualized very easily understandable way as should in Figure 1.

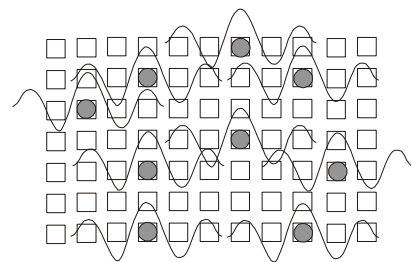


Figure 1: Neuron grid with activity centres

To create during the learning phase a complex activity pattern a neighbourhood function like a Sinc function with rotational symmetry or a Gaussian function have to be used.

The Sinc function, also known as Mexican hat function, corresponds to the Fourier Transform of the 'Tophat' function, which is in turn really just the 'Pupil' function. A multiplication in Fourier space by a Pupil function corresponds to band-limiting an input activation through an aperture, truncating the corners; in direct space, the convolution of the input pattern against the Fourier Transform of the kernel does the same. In short, the Sinc function is

$$\text{Sin}(r)/r \text{ [}=1 \text{ at } r=0\text{]}$$

as shown in Figure 2 and the convolution kernel can be approximated in matrix form by $\{\{1,-2,1\},\{-2,4,-2\},\{1,-2,1\}\}$, whereby it is the Fourier Transform of the Heaviside Function $H(r)$ for which holds $[H(r)=1 \text{ for } r \leq 1, 0 \text{ for } r > 1]$.

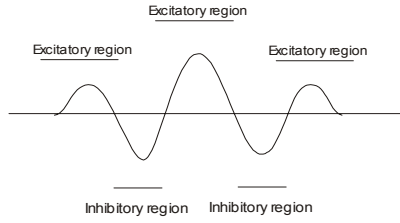


Figure 2: Shape of the used Sinc neighborhood function

Especially the existence of several excitatory and inhibitory regions of the Sinc function enables the an adequate way of the computing with activity patterns as now diverse effects of influences between different activity patterns can be taken into account. The number of these regions will be problem-oriented and has to be set before building the net. In the designing phase it also has to be decided, in which dimensions the neighborhood functions act, resp. if the net is regarded as a homogeneous grid on which the neighborhood function acts in a rotational symmetrical way or if special layer symmetries have to be taken in consideration. Furthermore the dimension of the Sinc function must be chosen in a way that the maximums and minimums meet neurons on the grid.

Because of these reasons, the activity function of a neuron should have the following form

$$\psi(\vec{q}, \vec{E}, k) = \frac{\vec{E}}{\vec{I}} \frac{\sin(d(\vec{q}, k))}{d(\vec{q}, k)}$$

Here $\psi(\vec{q}, \vec{E}, k)$ is a function of the position \vec{q} of the active neuron on the grid, a function of the intensity (vector) \vec{E} of the information to be classified, a function of the actual juncture k and a function of the distance d describing the influence of the Sinc function to the adjacent parts of the net.

The vector \vec{I} signifies a vector of standardization to meet our activity boundary of $[-1,1]$.

The given form of the neighborhood function $\psi(\vec{q}, \vec{E}, k)$ also guarantees that different activity patterns cannot be stored too close to each other, if the dimension of the Sinc function is chosen in that way that the minima and maxima meet the adjacent neurons as shown in Figure 1.

By this way the feature to separate independent activities will not be gone lost as two activity centres cannot be neighbored directly. Otherwise the neighborhood function guarantees that similar activity patterns are close enough to enable chunking effects by providing the positioning of so far not-chunked activities patterns.

For the shape of the envelope of $\psi(\vec{q}, \vec{E}, k)$ we assume the following formula

$$\text{Env}(\psi(\vec{q}, \vec{E}, k)) = e^{-\frac{d(\vec{q}, k)^2}{f^2}}$$

where f denotes a standardization parameter. As this envelope is nothing else then a Gaussssian function, it can be used unescorted if no chunking effects are requested.

For the given reasons for the activity function with its neighbourhood function holds

$$\psi(\vec{q}, \vec{E}, k) = \frac{\vec{E}}{\vec{I}} \frac{\sin(d(\vec{q}, k))}{d(\vec{q}, k)} e^{-\frac{d(\vec{q}, k)^2}{f^2}}$$

As every single neuron of the grid can act as a source of such an activity function, the global activity $\Psi(\vec{q}, \vec{E}, k)$ of the grid can be regarded as a superposition of the single activities $\psi_l(\vec{q}, \vec{E}, k)$, where l points out the single neuron.

The global activity pattern $\Psi(\vec{q}, \vec{E}, k)$ is given by the formula

$$\Psi(\vec{q}, \vec{E}, k) = \sum_{l=1}^L \psi_l(\vec{q}, \vec{E}, k) = \sum_{l=1}^L \frac{\vec{E}_l}{\vec{I}_l} \frac{\sin(d(\vec{q}_l, k))}{d(\vec{q}_l, k)} e^{-\frac{d(\vec{q}_l, k)^2}{f^2}}$$

In Figure 3 a trained SOM is shown, which used the Gaussian neighbourhood function with the conditions discussed above (referable the grid).

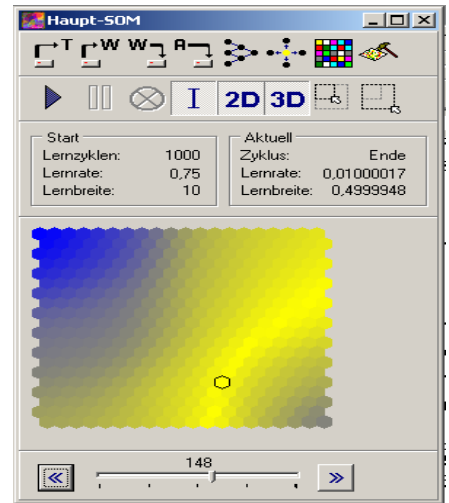


Figure 3:Classification result 1 of the trained SOM

As we can point out very easily the marked winner neuron is a part of the light global activity structure only and as we can understand easily too the change of this global structure must involve a change of the situation classified by the SOM even if the winner neuron is the same as before, as shown in Figure 4.

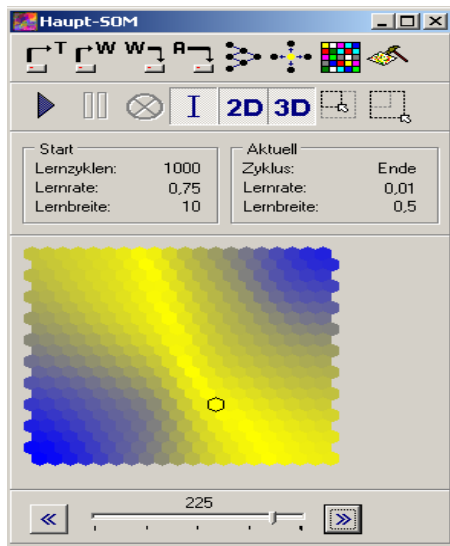


Figure 4: Classification result 2 of the trained SOM

Now we understand, why the interpretation of the global net activity bears a significant enhancement by the comparison to the common interpretation of the winner neuron only: much more situations (classification results) can be stored in the net.

Even as these result points out a essentially improvement of the capacity of storing classification results, another change of the structure of the SOMs must be taken into account, if we want to follow the context: "computing with activities". As we have discussed before the storing of activities requires special neighbourhood functions and owing to them a structure in which all neurons have equal rights regarding their neighbourhood. This special requirement demands that the structure of the SOMs follows an closed loop as shown in Figure 5, as otherwise marginal problems at the edges of the SOM will occur.

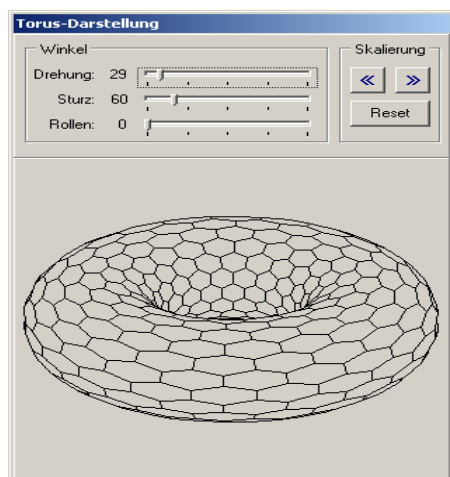


Figure 5: Topological form of a closed SOM

In Figure 6 the classification behaviour of the closed SOM is shown. Clearly it can be pointed out, that the activity pattern has a highly symmetrical form now, which can be interpreted in a better way than before. On such kinds of nets no marginal problems will occur.

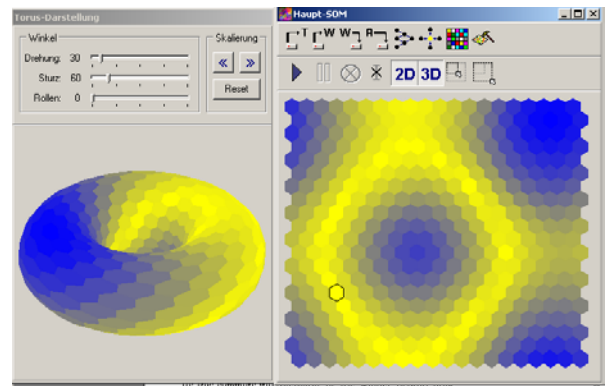


Figure 6: Classification result of a trained closed SOM

EXPERIMENTAL RESULTS

Figure 7 shows a first application of the theory discussed above. The categorization system was designed to analyze the scenes of soccer playing robots. The diverse situations of the game are categorized by a SOM of 100×150 neurons. The SOM is placed on a closed topology (torus) and was trained by 150000 training cycles whereat every training pattern represents a situation of a soccer match. As clearly can be pointed out from Figure 7 the SOM shows a very particular landscape of an activity pattern in which all possible situations of the match are coded. The winner region is placed in the lower left region. The light regions are the regions of high activities

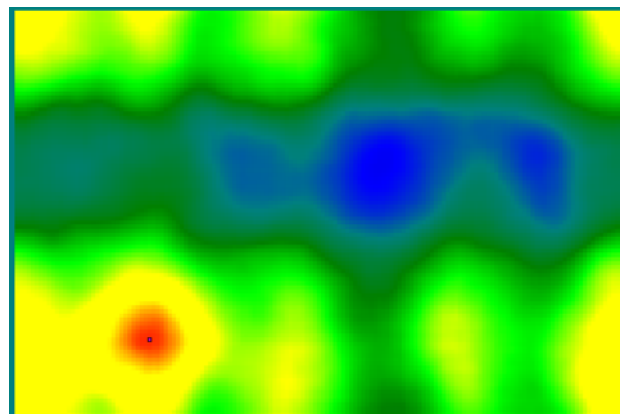


Figure 7: Structure of a SOM representing a soccer match

It is most important to mention, that the special form of coding is evaluated by the Som itself. That means that Figure 7 points out a highly individual neuro-mental representation of the situation which will be decoded for "us" only if we watch the reaction of the ruled robots. This is the same situation we observe in biological neural networks, as their coding can be decoded up to today also not in an adequate way. But at the end of the day about this fact we must not care in anyway, as a blackbox coding doesn't say anything about the power a neural net can and will act. In Figure 8 the interface of the simulation and classification program which involves the net structures discussed above is shown. On the upper left hand side of the figure the

momentary scene of a soccer game of two simulated robots is visualized. Thereunder the activity of the categorizing SOM is shown. At Figure the light area of the SOM represents the main point of the neuro mental representation of the momentary scene of the game. Please note that the scene is not characterized by the black dotted winner neuron – which is place in the left lower light region - but by the all around activity pattern of the closed SOM.

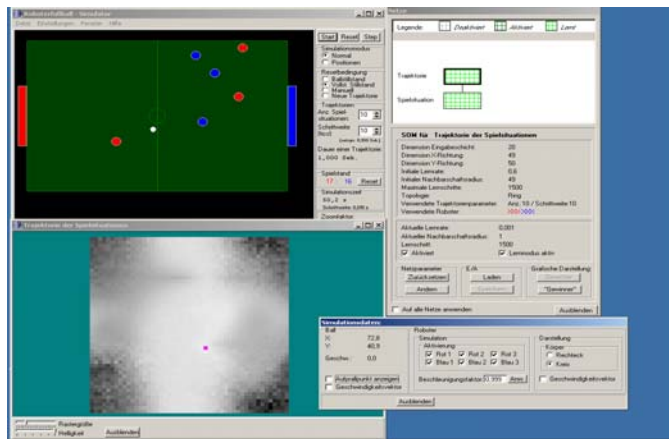


Figure 8: Interface of the simulation and classification programm

Figure 9 a-c shows the categorization of the trajectories of the play situations by a closed SOM of 48×50 neurons. This SOM is in front off the SOM of Figure 8 and represents a higher order of the neuro-mental processing by closed SOMs. At this a trajectory consists of 10 being related situations of the game, whereat the number “10” results out of out experiance. Surley it must be pointed out that normally the number of scenes which have to be combined to represent a whole move of the game must be situation-relevant. To find out such an adequate number a new “higher ordered” modules have to place in the hierarchy which determine the situations which have to be involved in move-representation. This kind of focus of attention is one of our major researching areas of the moment.

Once again in Figure 9a-d the bright activity patterns are the representative of the categorized input pattern which now is given by the structure of the $10 \times 100 \times 150$ values of activities from the SOM representing the situations of the game. The training cycles have been the same as for the SOM representing the situations.

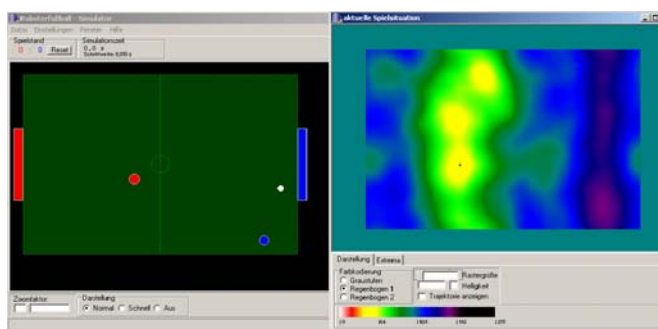


Figure 9a: Situation categorization of the scene of soccer playing robots by activity patterns of a “closed” SOM

The next Figures show the devolpement of the game within the early next time period. As clearly can be pointed out the major structure of the game does not change, but for the offender the situation becomes more and more interesting. This situation is code in the global activity structure in that way, that the light brigh of the SOMs is divided in two parts and the main point of the pattern is moving from the upper side to the lower side.

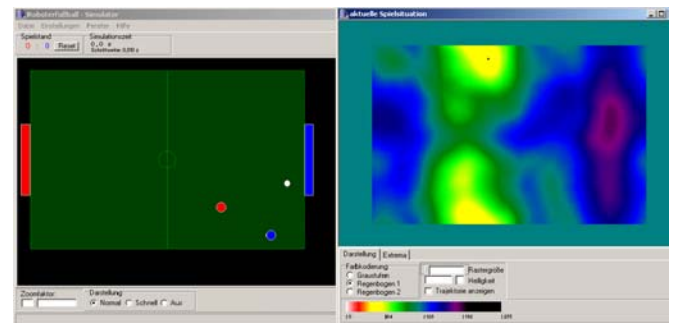


Figure 9b: Situation categorization of the scene of soccer playing robots by activity patterns of a “closed” SOM

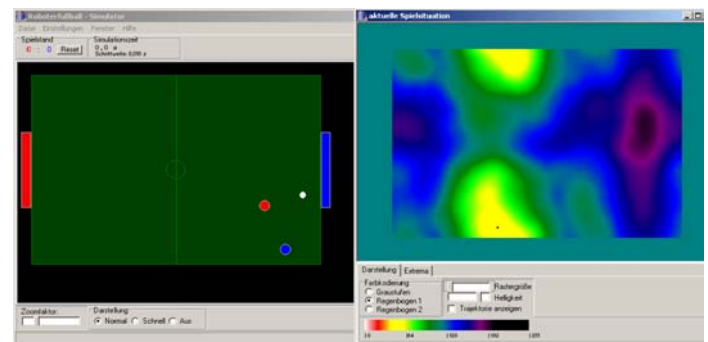


Figure 9c: Trajectory categorization of the scene of soccer playing robots by activity patterns of a “closed” SOM

That on the other side the global structure of the activity pattern is changing a little bit only affirms that our theoretical framwork holds in the case of application too.

Conclusions

To use the all around activity distribution of a neural neuron cluster to identify classification concepts may lead to a substantially new method to deal with neural nets. First time it seems to be possible to combine so far isolated stored classification concepts to high order classification structures by the method "computing with activities". As first experiences with SOM-structures on a torus show, by this way of classification/categorization it is possible to store all relevant features of a complex scene in the structure of the activity pattern of a closed SOM in such a way that each of the features of the scene is (subdominant) represented and still can be identified or interact with other originating features if the situation changes.

References

1. Aizenberg, I. N. 1999. „Neural Networks Based on Multi-valued and Universal Binary Neurons”. *LNCS 306-317*. Springer, Dortmund, 1625.
2. Alonso, E. (Ed). 2002. “Advances in Plan-Based Control of Robotic Agents”. *LNCS 2636*. Springer, Heidelberg.
3. Benesch, H. 1972. „Der Ursprung des Geistes“, dtv.
4. Birk, A. (Ed). 2002. “RoboCup 2001: Robot Soccer World Cup V”. *LNCS 2377*. Springer, Heidelberg.
5. Reuter, M. 1999. “About the Quantisation of the Neural Nets”. *LNCS 2101*. Springer, Dortmund, 530-542
6. Reuter, M. 2001. “Computing with, Activities”. *LNCS 2206*. Springer, Heidelberg, 174-184.
7. Suri, N.(Ed.) 2002. “Mobile Agents”. *LNCS 2535*, Springer, Heidelberg

MATTHIAS REUTER was born in Dresden, Germany and went to the Technical University of Clausthal, where he studied physics and geophysics. He obtained his doctor in 1987 and his PD 2002. He worked as scientist at the University of the German Army in Hamburg, the University of Dortmund and the Technical Universtiy of Clausthal. He deleoped a couple of new neural nets structures and learning strategies. His major investigations areas are neuronal net dynamics and robotics.

Multibody Applications

IDENTIFICATION OF DYNAMICAL PARAMETERS OF ROBOTIC MANIPULATORS USING SINGULAR VALUE DECOMPOSITION

Nader Nariman-Zadeh
Ahmad Bagheri
Mina Khoshnejad
Department of Mechanical Engineering
University of Guilan
Rasht, Iran
E-mail: nnzadeh@guilan.ac.ir

KEYWORDS

Singular Value Decomposition (SVD), Control, Identification, Robots

ABSTRACT

In this paper, Singular Value Decomposition (SVD) is used to identify the re-grouped dynamical parameters involved in the design of computed-torque control of a two degree of freedom direct-drive robotic manipulator. The effectiveness of using SVD in order to remove the singularities which may exist in the systems of equations during the identification process of such robotic manipulator is demonstrated by simulation results for three different trajectories in an on-line estimation process.

INTRODUCTION

The computed-torque method as a non-linear model-based controller (Craig 1988) is frequently used in the control of non-redundant robotic manipulators (Craig 1988) (Koivo 1989) (Porter et al. 1984) (Nariman-Zadeh and Darvizeh 2000). Dynamical behaviour of such robotic manipulators are governed by vector-matrix differential equations of the form

$$M(\theta)\ddot{\theta} + V(\theta, \dot{\theta}) + g(\theta) + f = \tau, \quad (1)$$

In this equation, $M(\theta) \in R^{n \times n}$ is the inertia matrix, $V(\theta, \dot{\theta}) \in R^n$ is the vector of centrifugal and coriolis torques, $f \in R^n$ is the vector of Coulomb friction torque, $\tau \in R^n$ is the vector of actuator torques, and $\theta \in R^n$ is the vector of joint angles. This equation can be represented for a planar two degree of freedom direct-drive robotic manipulator shown in figure (1) in a regrouped form as follows

$$M(\theta) = \begin{bmatrix} p_1 + 2p_3 \cos(\theta_2) & p_2 + p_3 \cos(\theta_2) \\ p_2 + p_3 \cos(\theta_2) & p_2 \end{bmatrix}, \quad (2a)$$

$$V(\theta, \dot{\theta}) = \begin{bmatrix} -\dot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2)p_3 \sin(\theta_2) \\ \dot{\theta}_1^2 p_3 \sin(\theta_2) \end{bmatrix}, \quad (2b)$$

$$f(\theta, \dot{\theta}) = \begin{bmatrix} P_4 \operatorname{sgn}(\dot{\theta}_1) \\ P_5 \operatorname{sgn}(\dot{\theta}_2) \end{bmatrix}. \quad (2c)$$

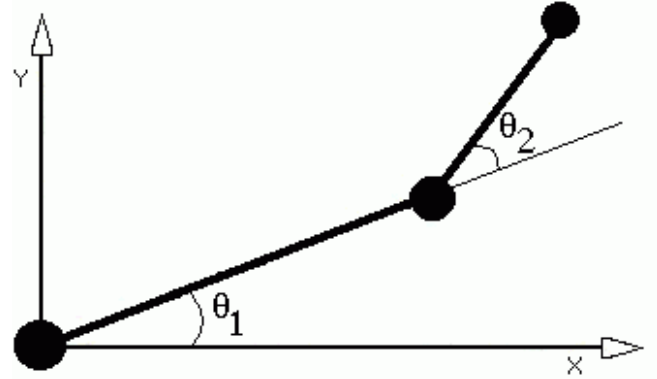


Figure 1: A Planar two Degree of freedom Robot Manipulator

If the dynamical parameters of such manipulators are precisely known, it is then possible to introduce the computed-torque control law in the equations of the form

$$\tau = M(\theta)(\ddot{\theta}_d + K_d \dot{e} + K_p e) + h(\theta, \dot{\theta}), \quad (3)$$

In equation (3), $\theta_d \in R^n$ is the vector of desired joint angles, $e = \theta_d - \theta \in R^n$ is the vector of joint angle errors, $K_p, K_d \in R^{n \times n}$ are appropriate gain matrices, and

$$h(\theta, \dot{\theta}) = V(\theta, \dot{\theta}) + g(\theta) + f. \quad (4)$$

Applying the control law (3), it is therefore evident from both equations (1) and (4) that the vector of joint-angle errors is governed by the vector-matrix differential equation

$$\ddot{e} + K_d \dot{e} + K_p e = 0, \quad (5)$$

This equation clearly indicates that gain matrices $K_p \in R^{n \times n}$ and $K_d \in R^{n \times n}$ can be appropriately selected so as to

produce second-order dynamics with any required characteristics. For example, by choosing these gain matrices in diagonal form, the error dynamics of each joint of the robotic manipulators are decoupled. The dynamical parameters of such robotic manipulators, however, are not known precisely in practice. Thus, such computed-torque control law will not have the form of equation (3) but rather the form

$$\tau = \hat{M}(\theta)(\ddot{\theta}_d + K_d \dot{e} + K_p e) + \hat{h}(\theta, \dot{\theta}) \quad (6)$$

where $\hat{M}(\theta) \in R^{n \times n}$ is the estimated value of the inertia matrix and $\hat{h}(\theta, \dot{\theta}) \in R^n$ is the estimated value of centrifugal, coriolis, gravitational and frictional vector defined in equation (3). In such cases, it is now evident that the vector of joints error dynamics is governed by the vector-matrix differential equation,

$$\ddot{e} + K_d \dot{e} + K_p e = \hat{M}^{-1}(\theta) \left[\{h(\theta, \dot{\theta}) - \hat{h}(\theta, \dot{\theta})\} + \{M(\theta) - \hat{M}(\theta)\} \ddot{\theta} \right] \quad (7)$$

It is evident that equation (7) reduces to equation (5) if $\hat{M}(\theta) = M(\theta)$ and $\hat{h}(\theta, \dot{\theta}) = h(\theta, \dot{\theta})$. However, in general $\hat{M}(\theta) \neq M(\theta)$ and $\hat{h}(\theta, \dot{\theta}) \neq h(\theta, \dot{\theta})$ and the choice of gain matrices will not normally be sufficient to ensure that the error dynamics of the robotic manipulator are acceptable. Therefore explicit identification schemes have been introduced in combination with computed-torque controllers in order to provide good estimates of $M(\theta)$ and $h(\theta)$ (Craig 1988).

ON-LINE IDENTIFICATION OF DYNAMICAL PARAMETERS USING SVD

In the case of the two-link direct-drive robotic manipulator, the governing vector matrix differential equations (1) can be alternatively expressed in form

$$\phi(\theta, \dot{\theta}, \ddot{\theta})P = \tau \quad (8)$$

where

$$\phi(\theta, \dot{\theta}, \ddot{\theta}) = \begin{bmatrix} \ddot{\theta}_1 & \ddot{\theta}_2 \\ 0 & \ddot{\theta}_1 + \ddot{\theta}_2 \\ (2\ddot{\theta}_1 + \ddot{\theta}_2)\cos\theta_2 - \ddot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2)\sin\theta_2 & \text{sgn}(\dot{\theta}_1) & 0 \\ \ddot{\theta}_1 \cos\theta_2 + \dot{\theta}_1^2 \sin\theta & 0 & \text{sgn}(\dot{\theta}_2) \end{bmatrix} \quad (9)$$

and

$$P = [p_1 \quad p_2 \quad p_3 \quad p_4 \quad p_5] \quad (10)$$

In equation (10), p_1, p_2 and p_3 are inertial and geometrical parameters whilst p_4 and p_5 are Coulomb friction parameters. In this re-grouped formulation, equation (7), which represents the error dynamics, will alternatively be given by

$$\ddot{e} + K_d \dot{e} + K_p e = \hat{M}^{-1}(\theta)\phi(\theta, \dot{\theta}, \ddot{\theta})(P - \hat{P}) \quad (11)$$

where \hat{P} is the estimated parametric vector defined in equation (10). It is evident that equation (11) reduces to equation (5) if $\hat{P} = P$.

In order to have high-accuracy computed-torque controllers, it is clear from equation (11) that, the parametric vector P has to be properly identified. In this way SVD can be conveniently used to solve the system of equations (11) on-line, in order to identify such parametric vector to overcome any possible singularities. The actual parametric vector, $P = [3.316 \quad 0.116 \quad 0.161 \quad 5.3 \quad 1.1]$ has been used as the actual dynamic of the robot (in matrices $M(\theta)$ & $h(\theta, \dot{\theta})$), where initially-guessed parametric vector, $\hat{P} = [1.5 \quad 0.05 \quad 0.05 \quad 0.2 \quad 0.05]$ has been used in the computed-torque control law (in matrices $\hat{M}(\theta)$ & $\hat{h}(\theta, \dot{\theta})$) (Nariman-Zadeh and Darvizeh 2000)

This means that the actual vector of dynamical parameters P has been used as the model in the simulation whilst the initially-guessed vector of dynamical parameters \hat{P} has been used to implement the computed-torque control method in order to identify the actual vector of dynamical parameters P . Therefore, equation (11) can now be solved optimally for $(P - \hat{P})$ (The difference between the actual dynamical parameters and the initially guessed ones) using SVD by collecting or computing a time-dependant set of kinematics signals $\theta(t)$, $\dot{\theta}(t)$, $\ddot{\theta}(t)$, $e(t)$, $\dot{e}(t)$ and $\ddot{e}(t)$ according to a desired trajectory of each joint of the robotic manipulators. Such collection of kinematics data can be accomplished in a finite number of sample times, which would be at least 5, according to the number of the unknown re-grouped dynamical parameters of the planar two-degree of freedom robotic manipulator. It is now evident that, for example, a coefficient matrix of unknown parameters would have the size of 10×5 if the kinematics data were collected in 10 successive sample time. Thus, the use of SVD to solve such linear systems of equation (11) for $(P - \hat{P})$ becomes more significant and necessary if there exist possible singularities in such collected data.

In fact, SVD is the method for solving most linear least squares problems that some singularities may exist in the normal equations. The SVD of a matrix, $A \in R^{n \times 5}$, where $A = \hat{M}^{-1}(\theta)\phi(\theta, \dot{\theta}, \ddot{\theta})$ and n is the number of data sample for identification process, is a factorization of the matrix into the product of three matrices, a column-orthogonal matrix $U \in R^{n \times 5}$, a diagonal matrix $W \in R^{5 \times 5}$ with non-negative elements (singular values), and an orthogonal matrix $V \in R^{5 \times 5}$ such that

$$A = U.W.V^T \quad (12)$$

$$A(P - \hat{P}) = Y \quad (13)$$

where Y is the vector of error dynamics, that is $Y = \ddot{e} + K_d \dot{e} + K_p e$. Consequently, optimal $(P - \hat{P})$ is calculated using the following relation

$$P - \hat{P} = V \cdot \left[\text{diag}\left(\frac{1}{w_j}\right) \right] U^T \cdot Y \quad (14)$$

using the modified inversion of matrix W (Press et al. 1985).

ILLUSTRATIVE EXAMPLES

In order to demonstrate the effectiveness of applying SVD on-line to identify the dynamical parameters in a computed-torque control schemes, three different joint-space trajectories have been used. In this way, SVD identification method has been only switched on 3 seconds after starting the motion. This is to show the effect of such identification of the dynamical parameters and demonstrating the improving performance of a model-based controller such as computed-torque control method. In the first trajectory, it is assumed that $\theta_{1d}(t) = \sin(t)$ and $\theta_{2d}(t) = \cos(t)$. The error dynamics together with position of each joint are shown in figures (2-5). It is very evident from figures (2-3) that SVD identification method quickly decreases the error dynamics of both joints to zero. It is also evident from figures (4-5) that very good trajectory tracking behaviour obtained shortly after 3 seconds when such SVD identification method started.

Similarly, figures (6-9) demonstrate same behaviour when the robotic manipulator is subjected to the second trajectory, $\theta_{1d} = 0.7\sin(2t)(1 - e^{-2t^2})$ and $\theta_{2d} = 1.2\sin(2t)(1 - e^{-2t^2})$.

Finally, figures (10-13) depict same behaviour for the third trajectory $\theta_{1d} = \sin(t) - t$ and $\theta_{2d} = \sin(t) - t$. The workspace trajectory tracking behaviour is also shown in figure (14) for the third trajectory.

It is very clear from these figures that computed-torque control method in conjunction with such SVD identification scheme presented in this paper can be successfully applied on-line for very good trajectory tracking control of robotic manipulators.

CONCLUSION

SVD has been successfully used in conjunction with computed-torque controllers for identification of dynamical parameters of practical direct-drive two-link robotic manipulators in an on-line estimation process. The results presented in this paper have demonstrated that SVD can simultaneously identify and improve the trajectory tracking behaviour of a model-based control scheme such as computed-torque method.

REFERENCES

- Craig, J.J. 1988. *Introduction to Robotics*. Addison-Wesley.
- Craig J.J. 1988. *Adaptive Control of Mechanical Manipulators*. Addison-Wesley.
- Koivo, A.J. 1989. *Fundamentals for Control of Robotic manipulators*. John Wiley.
- Lewis, F.L. 1996. "Neural Network Control of Robotic Manipulators". *IEEE Expert*, (June), 64-75.
- Nariman-Zadeh, N.; M. Darvizeh. 2000. "Genetic Identification of Dynamic Parameters in Computed-Torque Controllers for Robotic Manipulator". *International scientific conference, 70th Anniversary of Astrakhan State technology University*. (Russia, Apr), 24-27.
- Porter, B.; B. A. Sangolola.; and N. Nariman-Zadeh. 1984. "Genetic design of computed-torque controllers for Robotic Manipulators". *Proceeding IASTED international Conference on Systems and Controls*. (Lugano, Switzerland, June).
- Press, W. h.; S. A. Teukolsky; W. T. Vetterliny; and B. P. Flannery. 1992. "Numerical Recipes: The Art of Scientific Computing". 2th edition, Cambridge University Press.
- Zhihua, Q.; D. M. Dawson; J. F. Dorsey; and J. D. Duffie. "Robust estimation and control of robotic manipulators." 1995. *Robotica*, Vol. 13, 223-231

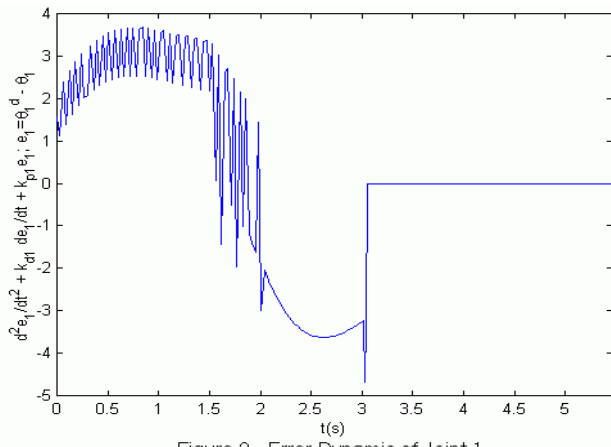


Figure 2 - Error Dynamic of Joint 1

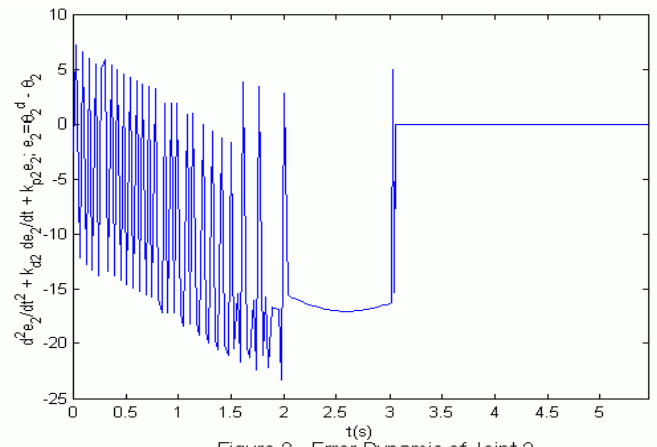


Figure 3 - Error Dynamic of Joint 2

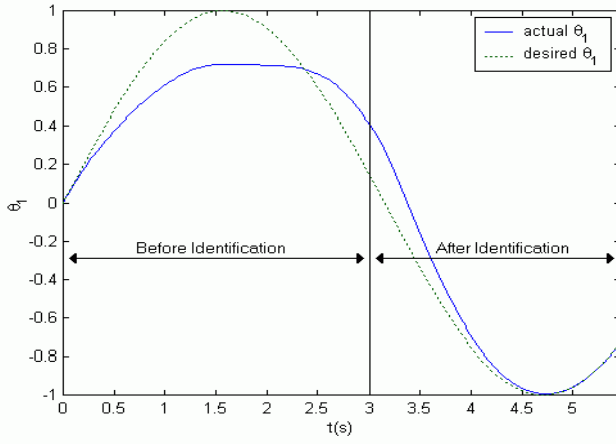


Figure 4 - Position tracking of Joint 1 for the first Trajectory

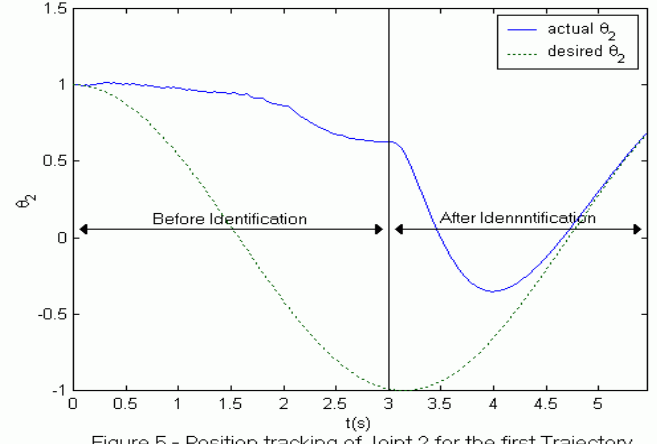


Figure 5 - Position tracking of Joint 2 for the first Trajectory

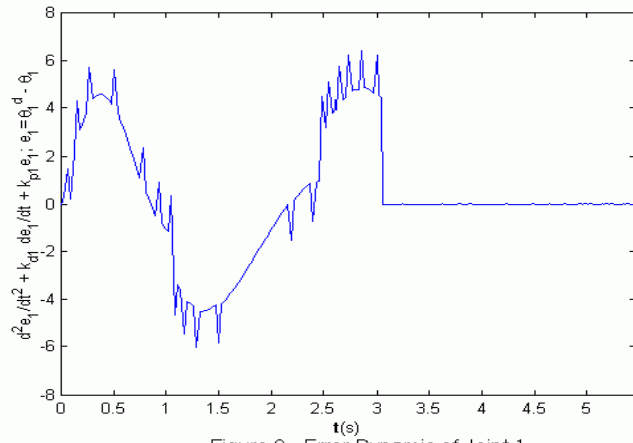


Figure 6 - Error Dynamic of Joint 1

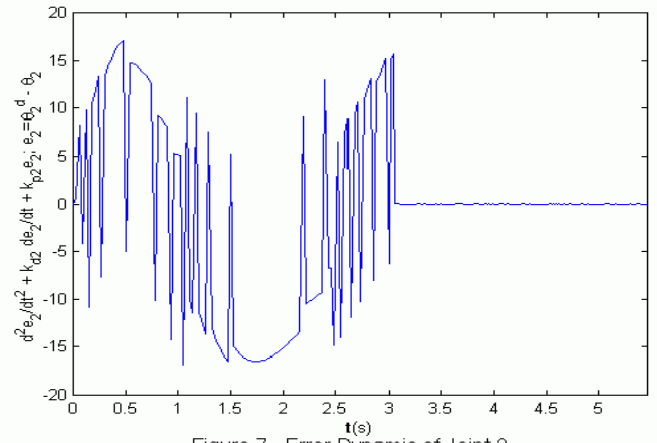


Figure 7 - Error Dynamic of Joint 2

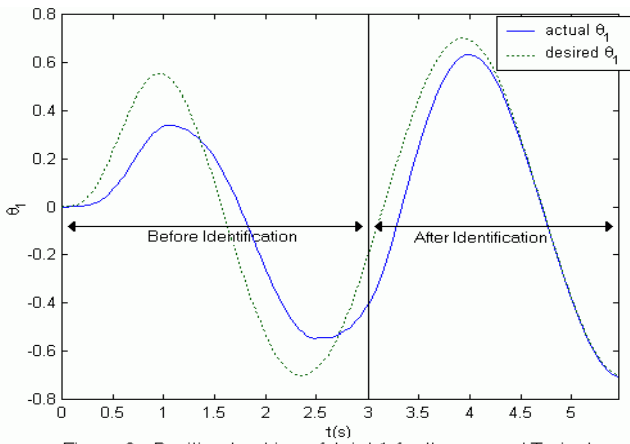


Figure 8 - Position tracking of Joint 1 for the second Trajectory

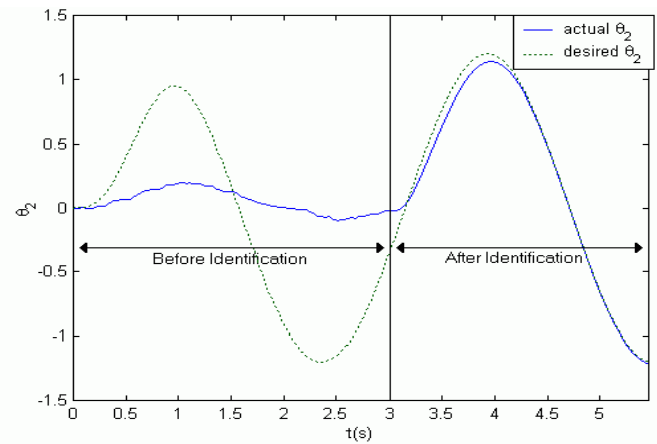


Figure 9 - Position tracking of Joint 2 for the second Trajectory

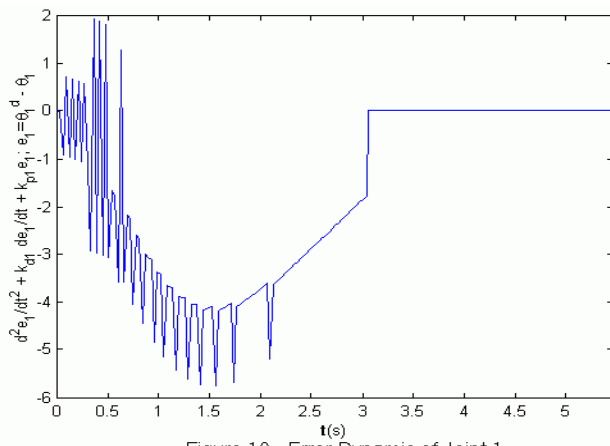


Figure 10 - Error Dynamic of Joint 1

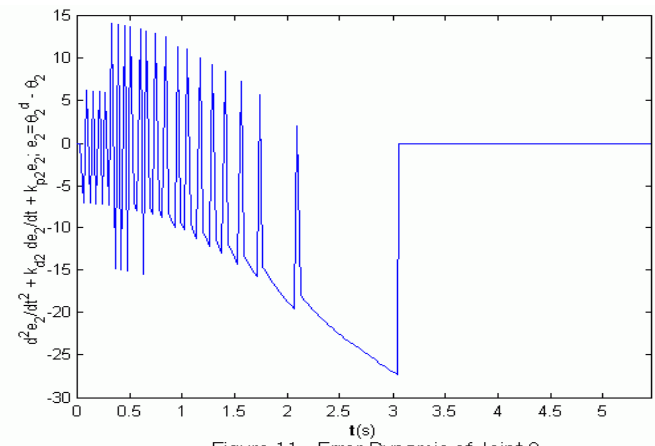


Figure 11 - Error Dynamic of Joint 2

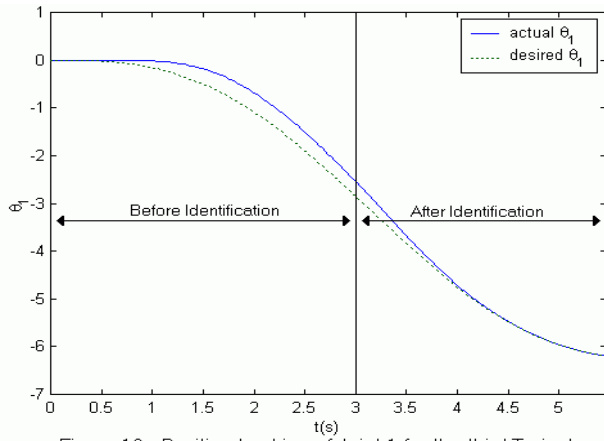


Figure 12 - Position tracking of Joint 1 for the third Trajectory

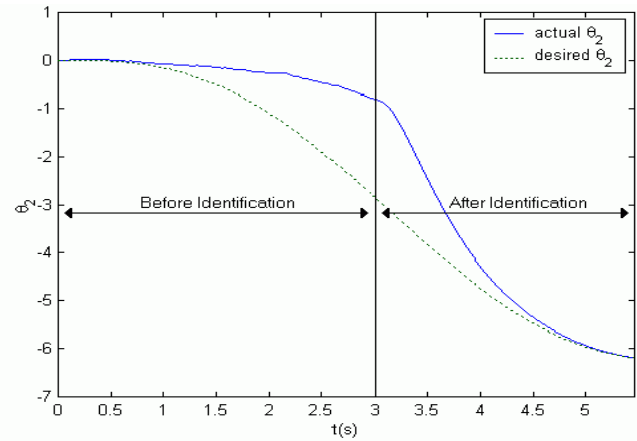


Figure 13 - Position tracking of Joint 2 for the third Trajectory

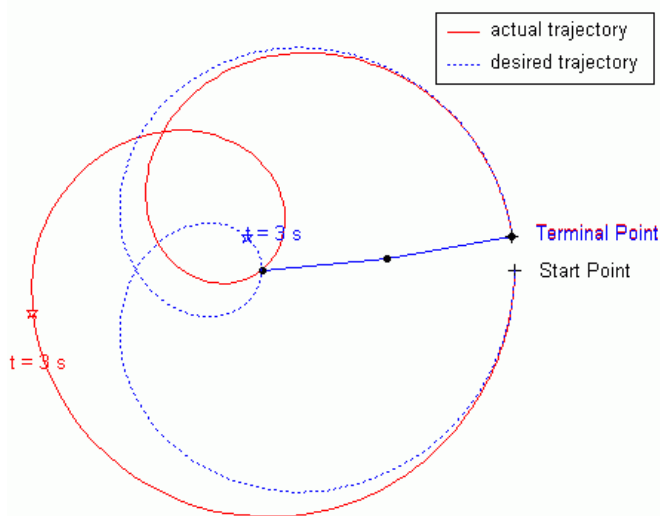


Figure 14 - Workspace Trajectory tracking for the third Trajectory

HYBRID CONTROL FOR A SNAKELIKE ROBOT

Viorel Stoian
University of Craiova,
Automation and Mechatronics Department
A.I. Cuza Street No. 13, 1100, Craiova, Romania
E-mail: stoian@robotics.ucv.ro

KEYWORDS

hyper redundant robot, position/force control, simulation of creeping and climbing locomotion.

ABSTRACT

This paper presents a new control algorithm for a hyper redundant manipulator, which moves in a labyrinth, and it can touch the boundaries of the niche. The collisions or contacts are a part of the task. The support points from the walls help to the moving process. A study of the hyper-redundant locomotion by creeping and climbing is achieved. The dynamic model by Lagrange method of the hyper-redundant robotic structure is established. A hybrid position/force controller for the hyper-redundant locomotion is proposed. Simulations of the hyper-redundant locomotion by climbing are presented.

INTRODUCTION

A Hyper Redundant Manipulator (HRM) has the potential ability to perform non-conventional tasks, such as moving in highly constrained environment or grasping a various sizes and shapes of objects (Hirose 1993).

The term hyper-redundant refers to redundant manipulators with a very large or infinite number of degree of freedom. These manipulators are analogous in morphology to snakes, elephant trunks, and tentacles (Chirikjian 1992). Is used the term "wave" to describe the mechanism deformations associated with these hyper-redundant locomotion gaits (repetitive sequences of mechanism deformations that cause net motion of the robot). Two highly idealized forms of biological locomotion are stationary wave locomotion and traveling wave locomotion.

The HRMs are categorized into 3 types of mechanisms: a large number of rigid links in series (a chain of discrete links) (Kobayashi and Ohtake 1995; Mochiyama et al. 1996; Paljug et al. 1995) etc., a cascade of modules composes of redundantly actuated parallel rigid platforms (Chirikjian and Burdick 1995; Zanganeh and Angeles 1995) and truly flexible physical structures tentacle-like (Chirikjian and Burdick 1995; Ivanescu and Stoian 1998).

This paper presents a serial rigid link robot manipulator with hyper degrees of freedom.

One of the cardinal goals of the robotic researchers is to provide control algorithms that allow robots to move in an unknown environment without collisions with the unstructured obstacles. Generally, in specific literature this problem is addressed as the collision avoidance problem (Chirikjian and Burdick 1992; Colbaugh et al. 1989; Khatib 1986; Zghal 1982). In this paper the hyper redundant manipulator is analyzed in following situation: his job is to infiltrate, to creep through dropped walls after a cataclysm in a humanitarian action or for inspection in highly constrained environments.

The mechanics are for a snake-like structure but with serial rigid links. In this case is necessary a compliant motion by touching of objects in many points of the manipulator like a human been in a very narrow cave which press with his shoulder the wall for his stability and with his hand push up a block (gravestone, manhole) or for climbing. The collisions or contacts are a part of the task.

The support points from the walls help to the moving process. For example, in Figure 1 (for simplicity, a planar structure) the robot fixes the i -link on the walls pressing the link's ends by F_i and F_{i+1} forces. In this situation the upper part (links $i+1, i+2, \dots, n$) can be relaxed and inferior part (links $1, 2, \dots, i-1$) can be constricted. This process allows a creeping in horizontal plane or a climbing in vertical plane.

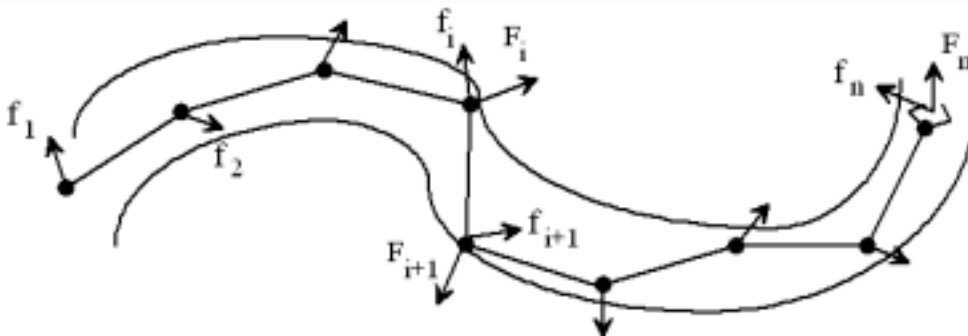


Figure 1: The HRM in a Labyrinth

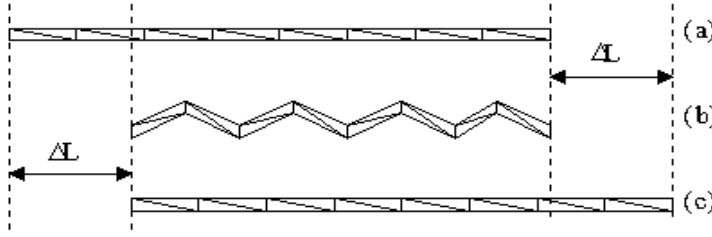


Figure 2. The hyper-redundant locomotion

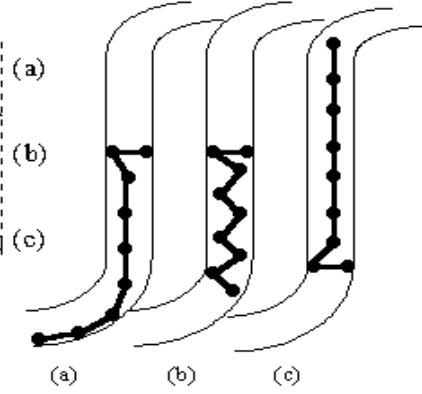


Figure 3: The Climbing Process

HYPER-REDUNDANT LOCOMOTION

Into a spatial and complex labyrinth is necessary to combine two kinds of actions: creeping and climbing. These actions, called “the hyper-redundant locomotion” are defined to mean motions resulting in net displacements of hyper-redundant robots due to the internally induced bending of the mechanism (Chirikjian and Burdick 1991). For simplicity, in this paper is studied a planar structure, but the ideas are generalizable to the spatial case. Figure 2 presents the hyper-redundant locomotion with amplitude varying wave that remains stationary with respect to body coordinates. ΔL represents the step for one cycle of motion. In (a) the robot there is in the initial position, in (b) it has the upper end strongly fixed in terrain, the inferior end is discharged and it moves ahead. In (c) the inferior end is strongly fixed in terrain end the upper end is discharged. Now, the hyper-redundant robot can relax ahead and it has again the initial position, but moved with the displacement ΔL .

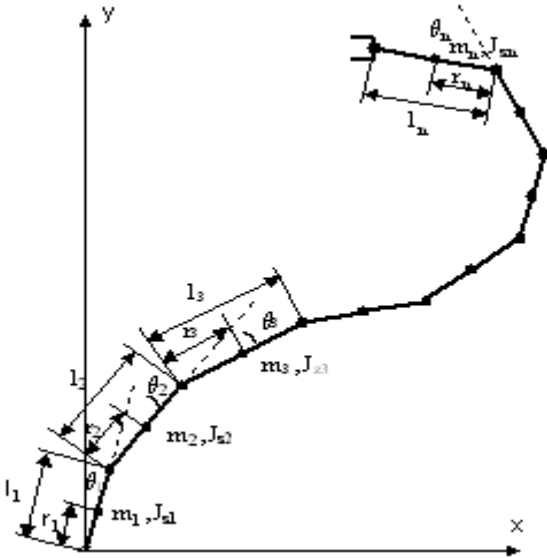


Figure 4: The HRM Structure

(Chirikjian and Burdick 1991) present the hyper-redundant locomotion with traveling wave with constant amplitude.

The two ends of the robot have the same functions like the ones from Figures 2. Here there is only one wave that travels the length of the body from the bottom of the mechanism to the upper end.

Figure 3 presents a climbing process by the same hyper-redundant robotic structure.

DYNAMIC MODEL

Figure 4 presents a hyper-redundant robotic structure for establish a dynamic model by Lagrange method. With the notations from that figure are passed get through all phases of the algorithm. The coordinates $x(q)$ and $y(q)$ are:

$$\begin{aligned} x_k &= \sum_{i=1}^n l_i \cos \left(\sum_{j=1}^i \theta_j \right); \\ y_k &= \sum_{i=1}^n l_i \sin \left(\sum_{j=1}^i \theta_j \right) \end{aligned} \quad (1)$$

The velocities of the x and y are:

$$\begin{aligned} \dot{x}_k &= - \sum_{i=1}^k \left\{ l_i \left[\sin \left(\sum_{j=1}^i \theta_j \right) \right] \left[\sum_{j=i}^k \dot{\theta}_j \right] \right\}; \\ \dot{y}_k &= \sum_{i=1}^k \left\{ l_i \left[\cos \left(\sum_{j=1}^i \theta_j \right) \right] \left[\sum_{j=i}^k \dot{\theta}_j \right] \right\} \end{aligned} \quad (2)$$

The kinetic energy and the potential energy are:

$$\begin{aligned} E_C(q, \dot{q}) &= \frac{1}{2} \dot{q}^T M(q) \dot{q}; \\ E_P(q) &= \sum_{i=1}^n E_{Pi} = g \sum_{i=1}^n m_i \left[\sum_{j=1}^i l_j \sin \left(\sum_{k=1}^j \theta_k \right) \right] \end{aligned} \quad (3)$$

where $M(q)$ is the inertial matrix and $q = [\theta_1, \theta_2, \dots, \theta_n]^T$

The Lagrange function is:

$$L(q, \dot{q}) = E_C(q, \dot{q}) - E_P(q) = \frac{1}{2} \sum_{i,j=1}^n M_{ij}(q) \dot{q}_i \dot{q}_j - E_P(q) \quad (4)$$

Now, we can apply Lagrange's equation:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = T_i \quad (5)$$

The two terms of the Lagrange's equation are:

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) &= \frac{d}{dt} \left(\sum_{j=1}^n M_{ij} \dot{q}_j \right) = \sum_{j=1}^n (M_{ij} \ddot{q}_j + \dot{M}_{ij} \dot{q}_j); \\ \frac{\partial L}{\partial q_i} &= \frac{1}{2} \sum_{j,k=1}^n \frac{\partial M_{kj}}{\partial q_i} \dot{q}_k \dot{q}_j - \frac{\partial E_P}{\partial q_i} \end{aligned} \quad (6)$$

Now, the Lagrange's equation is:

$$\sum_{j=1}^n M_{ij}(q) \ddot{q}_j + \sum_{j,k=1}^n \Gamma_{ijk} \dot{q}_j \dot{q}_k + \frac{\partial E_P}{\partial q_i}(q) = T_i \quad (7)$$

where :

$$\Gamma_{ijk} = \frac{1}{2} \left(\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} + \frac{\partial M_{kj}(q)}{\partial q_i} \right) \quad (8)$$

If we define the Coriolis matrix as:

$$\begin{aligned} C_{ij}(q, \dot{q}) &= \sum_{k=1}^n \Gamma_{ijk} \dot{q}_k = \\ &= \frac{1}{2} \sum_{k=1}^n \left(\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} + \frac{\partial M_{kj}(q)}{\partial q_i} \right) \dot{q}_k \end{aligned} \quad (9)$$

then $C(q, \dot{q})\dot{q}$ is the Coriolis and centrifugal forces vector. The third term is a position term representing loading due to gravity.

$$\frac{\partial E_P}{\partial q_i}(q) = G(q) \quad (10)$$

T_i represents the outer forces:

$$T_i = T_i - B(\dot{q}) - \tau \quad (11)$$

where T_i are the actuators forces (inner generalized forces), B represents the friction forces and τ represents the forces from joints due to the forces acting on the tool tip or on the another points from robotic structure due to a load or contact with the environment. With the above-mentioned notations we have the dynamic model of the hyper-redundant robotic structure from Figure 4:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B(\dot{q}) + G(q) + \tau = T \quad (12)$$

In this paper, by "position" we mean "position and orientation", and by "force" we mean "force and torque".

HYBRID CONTROL

During creeping in tunnels or labyrinths the robot comes in contact with environment; therefore, interaction forces develop between the robot and environment. Consequently, these interaction forces, as well as the position of the points from the robot, must be controlled. A number of methods for obtaining force information exist: motor currents may be measured or programmed, motor output torques may be measured and joint-mounted sensors may be used. There are two primary methods for producing compliant motion: a passive mechanical compliance built into the manipulator, or an active compliance implemented in the software control loop, force control. Passive compliance offers some performance advantages, but the force control method offers the advantage of programmability. This allows the manipulator to use the particular form of compliance necessary for the particular application. Throughout this paper is assumed that the desired velocity and force trajectories, which are commanded by the controllers, are consistent with the model of the environment (Lipkin and Duffy 1998). If this is not the case, it may be possible to modify the desired velocity and force trajectories to be consistent with the model of the environment. The compliant motion occurs when the position of a robot is constrained by the task. The so-called hybrid position/force controller (Chae et al. 1988; Raibert and Craig 1981) can be used for tracking position and force trajectories simultaneously. The basic concept of the hybrid position/force controller is to decouple the position and force/control problems into subtasks via a task space formulation because the task space formulation is valuable in determining which directions should be force or position controlled. Figure 5 presents a hybrid position/force controller.

$$\mathbf{x}_d = \begin{bmatrix} x_d \\ y_d \\ z_d \end{bmatrix}; \mathbf{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}; \Delta \mathbf{x} = \mathbf{x}_d - \mathbf{x};$$

$$\mathbf{F}_D = \begin{bmatrix} F_{xd} \\ F_{yd} \\ F_{zd} \end{bmatrix}; \mathbf{F} = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}; \Delta \mathbf{F} = \mathbf{F}_D - \mathbf{F} \quad (13)$$

$$\mathbf{q} = [q_1, q_2, \dots, q_n]^T; \mathbf{f} = [f_1, f_2, \dots, f_n]^T \quad (14)$$

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{S} = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix}$$

$$\text{where } S_i = \begin{cases} 1 & \text{for force control} \\ 0 & \text{for position control} \end{cases} \quad (15)$$

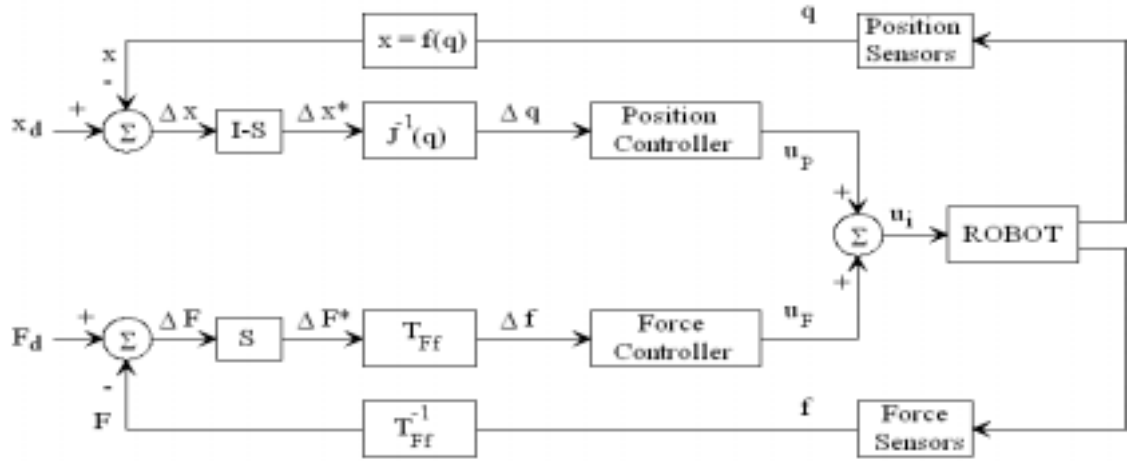


Figure 5: The Hybrid Position/Force Controller

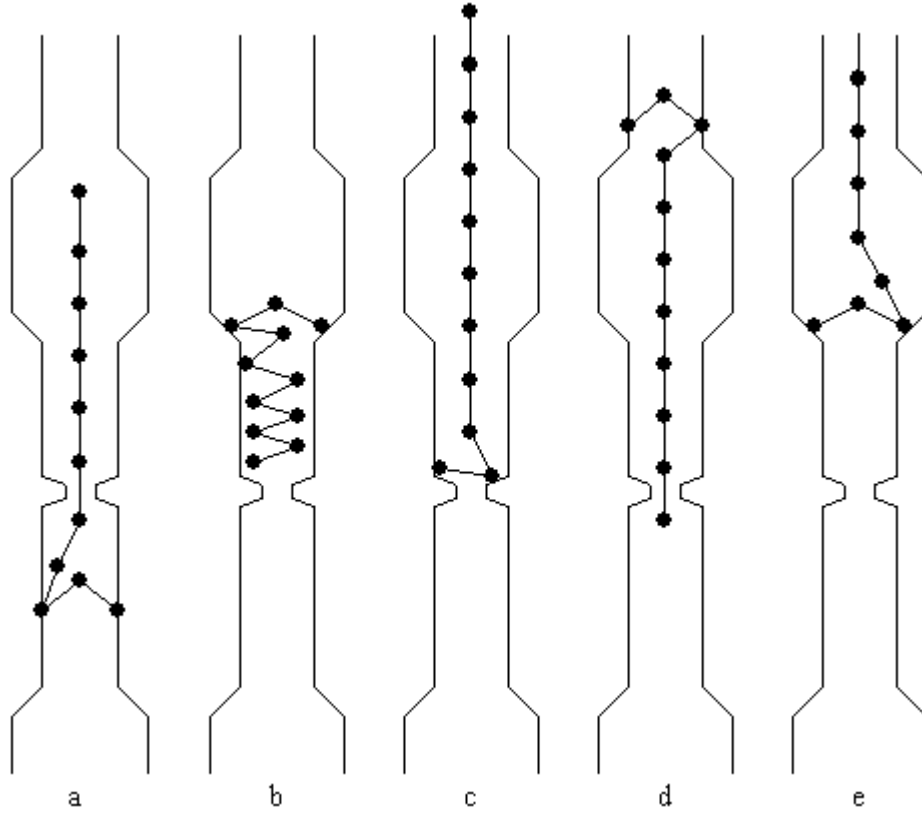


Figure 6: Simulation for a Climbing Action

S defines the compliance selection matrix.

$$T_{Ff} = J^T ; T_{Ff}^{-1} = (J^{-1})^T \quad (16)$$

where $J(q)$ is the Jacobian matrix and T_{Ff} is a matricial operator which make the transformation between the forces F and the forces f . $x = f(q)$ is a relation which represent the forward kinematics. For position controller is advantageously to use a PID controller and for force a PD controller:

$$u_p = k_1 \Delta q + k_2 \Delta \dot{q} + k_3 \int_0^t \Delta q dt \quad (17)$$

$$u_F = k_4 \Delta f + k_5 \Delta \dot{f} \quad (18)$$

The general control law u_i is:

$$u_i = \sum_j [\Psi_{ij} S_j \Delta f_i + \Xi_{ij} (1 - S_j) \Delta q_i] \quad (19)$$

where Ψ_{ij} is a force control law and Ξ_{ij} is a position control law.

Figure 6 presents a simulation for the climbing operation through a vertical labyrinth.

Any segments of the hyper-redundant robot are used for the anchorage of the structure on the walls of the vertical labyrinth while the others segments are moving up.

The hyper-redundant locomotion by creeping is analogous.

CONCLUSIONS

This paper presents a new control algorithm for a hyper redundant manipulator which moves in a labyrinth and it can touch the boundaries of the niche. The collisions or contacts are a part of the task. The support points from the walls help to the moving process. A study of the hyper-redundant locomotion by creeping and climbing is achieved. The dynamic model by Lagrange method of the hyper-redundant robotic structure is established. A hybrid position/force controller for the hyper-redundant locomotion is proposed. Simulations of the hyper-redundant locomotion by climbing are presented.

REFERENCES

- Chae, A.; C. Atkeson; and Hollerbach J. 1988. "Model-Based Control of a Robot Manipulator", Cambridge, MA, MIT Press.
- Chirikjian, G.S. and J.W. Burdick. 1991. "Kinematics of Hyper-Redundant Robot Locomotion with Applications to Grasping", *Proceedings of the 1991 IEEE International Conference on Robotics and Automation (ICRA'91)*, Sacramento, California, vol. 1, pp. 720-725.
- Chirikjian, G.S. and J.W. Burdick. 1992. "A Geometric Approach to Hyper-Redundant Manipulator Obstacle Avoidance", *Transactions of the ASME, Journal of Mechanical Design*, vol. 114, pp. 580-585.
- Chirikjian, G.S. and J.W. Burdick. 1995. "Kinematically Optimal Hyper-Redundant Manipulator Configurations", *IEEE Transactions on Robotics and Automat.*, vol.11, No.6, pp. 794-805.
- Colbaugh, R.; H. Seraji; and K.L. Glass. 1989. "Obstacle Avoidance for Redundant Robots Using Configuration Control", *J. of Robotic Systems*, vol. 6, pp. 721-744.
- Hirose, S. 1993. "Biologically Inspired Robots: Snake-like Locomotors and Manipulators", Oxford Univ. Press.
- Ivanescu, M. and V. Stoian. 1998. "A Control System for Cooperating Tentacle Robots", *Proceedings of the 1998 IEEE International Conference on Robotics and Automation (ICRA'98)*, Katholieke Universiteit, Leuven, Belgium, vol. 2, pp. 1540-1545.
- Kobayashi, H. and S. Ohtake, 1995. "Shape Control of Hyper-Redundant Manipulator", *Proceedings of the 1995 IEEE International Conference on Robotics and Automation, (ICRA'95)*, Nagoya, Japan, vol. 2, pp. 2803-2808.
- Khatib, O. 1986. "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", *Int. J. of Robotic Research*, vol. 5, pp. 90-98.
- Lipkin, H. and J. Duffy. 1988. "Hybrid Twist and Wrench Control for a Robot Manipulator", *Trans. ASME J. Mechan. Transmissions Autom. Design*, vol. 110, pp. 138-144.
- Mochiyama, H.; E. Shimemura; and H. Kobayashi. 1996. "Control of Serial Rigid Link Manipulators with Hyper Degrees of Freedom: Shape Control by a Homogeneously Decentralized Scheme and its Experiment", *Proceedings of the 1995 IEEE International Conference on Robotics and Automation (ICRA'96)*, Minneapolis, Minnesota, vol. 2, pp. 2877-2882.
- Paljug, E.; T. Ohm; and S. Hayati. 1995. "The JPL Serpentine Robot: a 12 DOF System for Inspection", *Proceedings of the 1995 IEEE International Conference on Robotics and Automation, (ICRA'95)*, Nagoya, Japan, vol. 3, pp. 3143-3148.
- Raibert, M. and J. Craig. 1981. "Hybrid Position/Force Control of Manipulators", *J. Dyn. Syst. Meas. Control*, vol. 102, pp. 126-133.
- Zanganeh, K. E. and J. Angeles. 1995. "The Inverse Kinematics of Hyper-Redundant Manipulators Using Splines", *Proceedings of the 1995 IEEE International Conference on Robotics and Automation (ICRA'95)*, Nagoya, Japan, vol. 2, pp. 2797-2802.
- Zghal, H.; R.V. Dubey; and J.A. Euler. 1992. "Collision Avoidance of a Multiple Degree of Redundancy Manipulator Operating Through a Window", *Transaction of the ASME, Journal of Dynamic Systems, Measur., and Control*, vol. 114, pp. 717-721.

AUTHOR BIOGRAPHY

VIOREL STOIAN was born in Segarcea, Romania and went to the University of Craiova where he studied automatic control and logic circuits. He obtained his degrees in 1983. After doing his military service, he worked at Faculty of Automatic Control, Computers and Electronics, Mecatronics department.

SIMULATION BASED EXPERIMENT DESIGN TO DIAGNOSE FAULTS OF RAILROAD WHEELSETS

Zofia Kowalska
Institute of Fundamental Technological Research PAS
ul. Świętokrzyska 21
00-049 Warsaw, Poland
E-mail: zkowal@ippt.gov.pl

KEYWORDS

Transportation; System dynamics; Nonlinear lumped parameter model; Parameter identification.

ABSTRACT

Simulations of dynamical interactions between the wheelset and the railway track are performed. The interest is focused on the dynamic response of the wheelsets passing over the intentional irregularities placed on rails. The aim is to confirm whether or not different wheelsets faults are possible to be reliably diagnosed on a special section of the track equipped with the normal force sensors at an array of points along the rails. The problems of modelling and simulation of dynamical interactions of bodies with irregularities on their rolling surfaces are discussed. Despite the fact that the relevant inverse problem is rather ill posed, and the measuring system would be expensive, useful information can be extracted by using appropriate procedures, which makes the whole idea promising.

INTRODUCTION

The objective of the paper is to present main aims and preliminary results of the research undertaken in order to design a reliable system, which would enable diagnostics of railroad wheelsets on a special railway track designed and equipped for testing. The signals of the normal forces between the wheels and the rails are traced at an array of discrete points along the rails and correlated with passing wheelsets.

The quantitative diagnostics, i.e. the estimation of the faults parameters from observations of the dynamical system response, can be classified into the inverse dynamical problems, which generally are ill posed. In experiment design for the least-squares parameter estimation sensitivity analysis of parameter estimates to observations may be helpful (Kowalska 1999). Other techniques of regularization, for example matrix singular values decomposition (SVD), are also recommended (Press et al. 1992). However, at the present stage of development of numerical tools for simulation and data analysis an approach based on simulations is even more useful and universal.

PROBLEM DESCRIPTION

Due to a complex dynamics of the train/track system diagnostic testing must be based on a reliable model of the system and simulation studies of its dynamical response to different faults. Of course, subsequent verification and calibration of the real system are necessary. The scheme of

simulation based experiment design for quantitative diagnostics is given in figure.1.

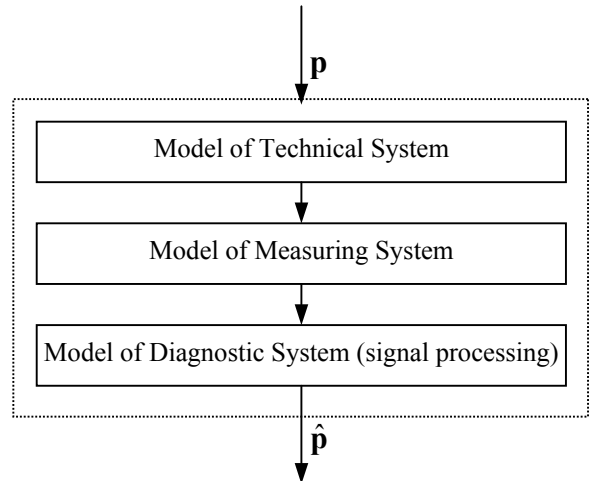


Figure. 1. Simulation Based Experiment Design

A basic function of the system is to measure the static axle load of passing wheelsets of a train. The more advanced function is to discern damaged wheelsets and general assessing a potential of rail vehicles to damage the track. The most difficult function to accomplish is to diagnose a type of faults and their quantitative, that is parametric, estimation. Different kinds of faults are of interests, like the wheels' misalignment, unbalance, tendency toward a sideways motion also known as hunting, and so called flats, resulting from over breaking. On the scheme in figure 2 the wheel profile has idealized extremely large flat spot.

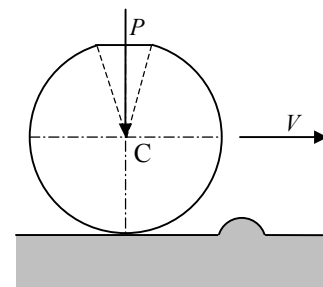


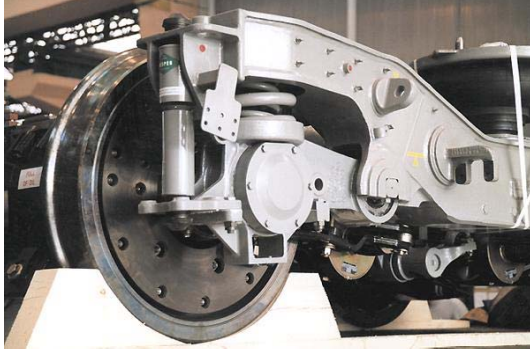
Figure 2: Flatted Wheel and an Intentional Irregularity on a Rail.

Inertial properties of wheelsets, compliance and damping of their primarily suspensions are of special interests. For illustration purposes the photo of the Alstom boogie primarily suspension of the wheelset are given in figure 3.

The basic diagnostic signal is the normal force between a wheel and a rail measured indirectly at discrete points along

the rail track. Simulation based on reliable models enables prediction of time patterns in the diagnostic signal that are characteristic for a specific fault.

One of the concepts is to place intentional irregularities on the rail in order to increase the distance of measured deterministic system response from unavoidable noise. (see figure 2).



Source: Alstom WWW Site

Figure 3. Photo of Alstom Boogie Primary Suspension

PHYSICAL AND NUMERICAL ASPECTS OF SIMULATION OF CONTACT VIBRO-IMPACTS

In order to explain main physical and numerical problems of modeling vibro-impacts of rolling bodies let us consider the simplest system of this type. It consists of a mass m rolling on a surface of the elastic half-space. The dynamics of adjacent sub-structures is neglected; only the inertia of a rolling mass and the contact compliance is taken into account. The irregularities are assumed to be very small in comparison to the rolling body radius.

If the rolling mass stays in contact with the foundation, and surface irregularities are very small in depth, the vertical motion of the mass centre in the inertial reference frame is governed by the equation:

$$m \frac{d^2 y_C}{dt^2} = -P + F(\delta) \quad (1)$$

Symbol y_C stands for vertical position of the body mass centre C . The external constant force $-P$ directed downward includes the weight of the rolling mass. The relationship $F(\delta)$ relates the normal force F upon the indentation δ , strictly speaking - the relative approach of remote points, outside the contact zone, in the contacting bodies. The relative approach may be expressed as

$$\delta = y_C - y_C^k$$

where y_C^k denotes the rigid constraints for the point C motion. In other words, the vertical kinematical position of the rigid body rolling on a rigid surface. Figure shows a cosine irregularity on the rail (line 1) and the relevant constraint (line 2) for smooth wheel (Bogacz and Kowalska 2001). The value of y_C^k is searched for numerically at every

step of numerical integration. If the rolling mass loses contact with the surface, the term $F(\delta)$ in Eq. 1 vanishes.

In accordance with the Hertz theory the relationship between the normal force F and the indentation δ has the form:

$$F(\delta) = C\delta^{3/2}$$

Herzian theory of elastic contact is well established, experimentally verified and very useful (Harris 1966), (Timoshenko and Goodier 1951). However, the theory has also significant limitations, resulting from its fundamental assumptions on shape of the bodies' surfaces and relative dimensions of a contact patch. The difficulties in modelling and simulations of dynamic contact of the rolling bodies with irregularities on circumference results from lack of reliable and useful models of non-Hertzian contact. At high speed, in the presence of deep irregularities with sharp edges plastic deformations occur (Kowalska and Bogacz 2002). They must be avoided for two reasons. The diagnostic system should not damage train gears in any way. In elastic-plastic range the model of wheel-rail contact is ambiguous, which may affect the diagnostics. Fortunately, according to Johnson's model of elastic-plastic contact of spherical bodies (Johnson 1985, Stronge 2000) final deformations and discrepancies of the force-indentation characteristic from the extrapolated Hertzian one are small for large curves and for the normal forces that are larger only 2-4 times than the limiting force for the first yield (Kowalska and Bogacz 2002). However, the irregularities introduced on rails on purpose should have shape and dimensions so that they could be modelled accurately at the present stage of knowledge. Moreover, the amplitudes of excited vibro-impacts should be limited.

High frequency vibro-impacts are difficult to simulate. Time histories are very sensitive to errors in times at which the bodies loose contact or come into contact. In general, searching for the instant contact points and the rigid constraint must be done numerically. For more than one possible unilateral constraints searching for active constraints is very time consuming, especially in two- and three-dimensional space.

The dynamical response depends on track properties. A stiff track lied on solid concrete blocks seems to be a proper solution because dynamical properties of such a system are relatively easy to control, model and stabilize.

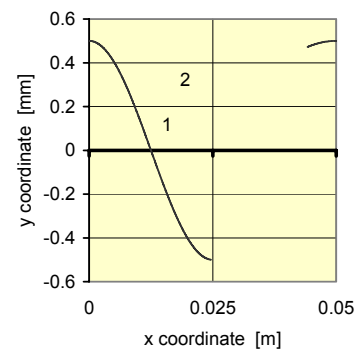


Figure 4. Rail Profile and Rigid Constraint

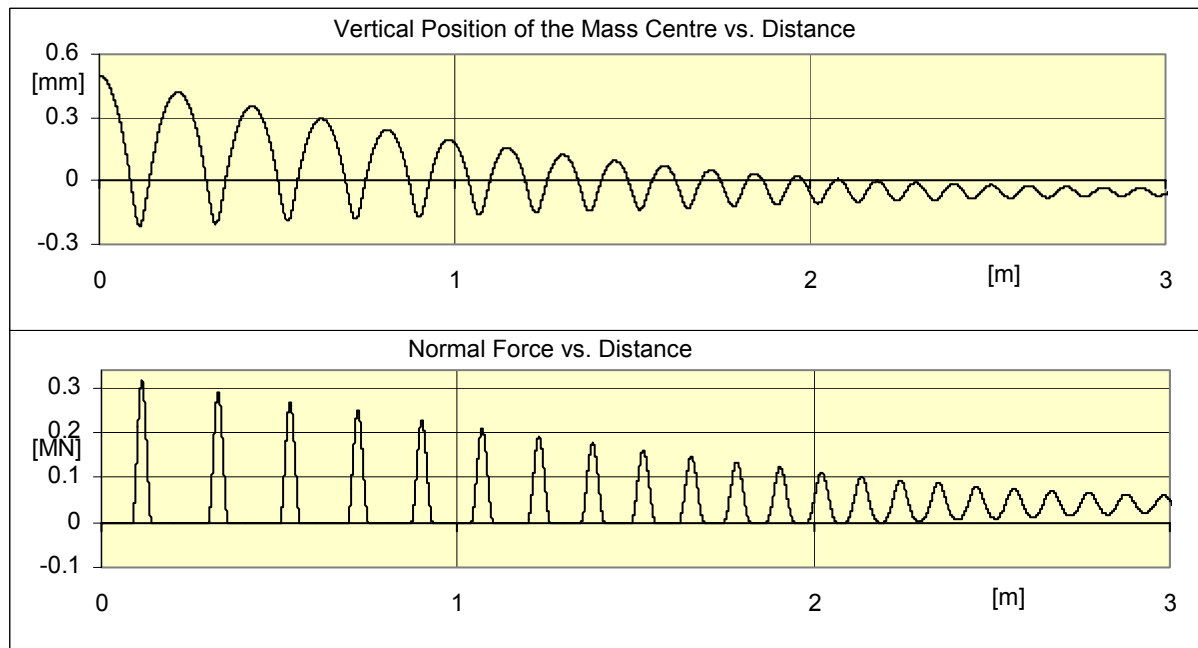


Figure 5. Transients of Moderately Damped Wheelset

Figure 5 illustrates the transient vibro-impact vertical motion of the wheelset. The motion is excited by two intentional identical irregularities placed on both rails.

In the case of impulse excitation with short irregularity on stiff track the dynamics of primarily suspension affects the response, mainly during the phase of vibro-impacts. Contact vibrations (without bounces) are less affected due to the fact, that the wheel/rail contact stiffness is much higher than the suspension stiffness.

The upper graph shows vertical position of the wheel center y_C versus the distance $l = V \cdot t$. The bottom graph presents the maximum values of the normal force F recorded along the rail, under the assumptions of constant train speed $V = 50$ km/h and continuous measurements both in time and space. If the maximum normal force is measured and recorded at discrete points of the rail, some information may be lost. The magnitude of normal forces and time periods between subsequent impacts can be used for diagnostics and parametric estimation of a wheelset mass, stiffness and damping of the wheelset primarily suspension.

Summarizing, in order to design proper test site oriented toward diagnostics of specific faults, it is necessary to: build the track of special properties; choose an appropriate diagnostic signal, that is, the response to particular excitation; find measurable features of the signal; develop procedures for diagnostics from the features.

The literature concerning the train/track or wheel/rail dynamic interaction and various related problems is enormous. Critical analysis of the whole literature would be a great effort and is beyond the scope of the paper. The reference section provides mainly the fundamental literature on Hertzian and Johnson's model of solid bodies contact. These models were converted into the computer models and used in simulation studies. Amongst others

references it is worth to mention the paper by Knothe (2000), which contains an excellent bibliography of tens related papers and the paper by Igeland and Ilias (1997), where the notion of the effective irregularity is introduced. The effective irregularity is analogous to the notion of the rigid constraint that is used herein, but they are not entirely equivalent. The system under consideration was also studied analytically with the approximate method of harmonic balance (Nayak 1972).

CONCLUSIONS

The results hitherto obtained show that the approach presented above to diagnose the wheelsets faults is technically difficult but may be useful after developing sophisticated procedures of collecting and processing measured data and subsequent calibration of the whole system.

Simulations show that intentional irregularities, properly designed and spaced on rails, are helpful in fault diagnostics of wheelsets, in particular - in diagnostics of the symmetry of mass distribution, damping and stiffness of the wheelset preliminary suspension.

In the case of many faults of interest, characteristic patterns in time histories of normal forces, which could be useful for diagnostics, occur only at relatively high train speed. That means that the length of the track must be large, hence the track itself and the measurements in general would be expensive. On the other hand, the advantages of diagnostic field tests of complete trains in comparison with laboratory tests of dismounted assemblies are obvious and therefore the efforts are worth to undertake.

REFERENCES

- Bogacz R. and Z. Kowalska. 2001. "Computer simulation of interaction between a wheel and corrugated rail." *European Journal of Mechanics A/Solids* 20, 673–684.
- Harris T.A. 1966. *Rolling Bearing Analysis* Wiley.
- Igeland A. and H. Ilias. "Rail head corrugation growth predictions based on non-linear high frequency vehicle/track interaction.", *WEAR* 213 (1997) 90-97.
- Knothe K. 2000. "Non-steady State Rolling Contact and Corrugations". In *Rolling contact phenomena 2000*, B. Jacobson and J.J. Kalker (Eds.). CISM Courses and Lectures No 411, Wien, Springer, 203–276.
- Johnson K.L., 1985. *Contact mechanics*, Cambridge University Press.
- Kowalska Z. 1999. "Sensitivity analysis in experiment design for identification of the electromechanical drive." In *Modelling and Simulation, Proc. of 13th European Simulation Multiconference ESM'99* (Warsaw, Poland, June 1-4). H. Szczerbicka (Ed.), Delft, Vol. 1, 491–496.
- Kowalska Z. and R. Bogacz. 2002. "Simulation of contact vibrations and impacts of compact hard bodies." *Machine Dynamics Problems* 25, No. 3/4, 97–106.
- Press, W.H.; S.A. Teukolsky; W.T. Vetterling; and B.P. Flannery. 1992. *Numerical recipes in Fortran*. Cambridge University Press.
- Nayak P.R. 1972. "Contact vibrations." *Journal of Sound & Vibration* 22, No. 1, 297–322.
- Stronge W. 2000. *Impact Mechanics*, Cambridge University Press.
- Timoshenko S. and J.N. Goodier. 1951. *Theory of elasticity*. McGraw-Hill.

Acknowledgement. This work was supported by the Polish State Committee for Scientific Research; project No. 5 T12C 019 24 granted from 2003-2005.

BIOGRAPHY

Zofia Kowalska was born in 1954 in Warsaw. In 1977 she received the M.Sc. degree from the Warsaw University of Technology, the Faculty of Precise Mechanical Engineering. Then she worked as a mechanical design engineer for the electronics company in Warsaw. Since 1983 she is with the Institute of Fundamental Technological Research (IFTR). In 1992 she received the Ph.D. degree in machine dynamics from the IFTR. Her major research interests include modelling and identification of different mechanical dynamical systems, recently - vibro-impact systems, in particular systems of rolling bodies with irregular surfaces.

NEW MODELS FOR FAST CONTACT FORCE COMPUTATION

Ignacio García
Marta Pla-Castells
Rafael J. Martínez
Instituto de Robótica
Universidad de Valencia
P.O. Box 2085
46071 Valencia, Spain

E-mail: [Ignacio.Garcia;Marta.Pla;Rafael.Martinez]@uv.es

KEYWORDS Dynamics, simulation, contact, friction, Coulomb Law.

ABSTRACT

This work focuses on the subject of impact response modeling in Real Time rigid body simulation, and more precisely in those applications where a very high accuracy in the solution is not required. The problem of impact response has several aspects to overcome, such as the instantaneous nature of the forces that appear during the contact, the absence of a unique solution in the case of contacts with friction, or the stiffness of the resultant models. This makes the impact management to be one of the most problematic parts to resolve during the implementation of computational models for rigid body systems. We present a model for contacts that is much simpler than the usual methods and that achieves an accuracy of the same order than the integration step size respect to the usual method. The model is valid both for frictional and non-frictional contacts.

We have performed some numerical experiments to validate the model which confirmed that one of the main advantages of the method versus the traditional ones is its simplicity and the lower computational cost it exhibits

INTRODUCTION

The problem of multi-body systems simulation has been extensively studied by many authors during the last decades, in the fields of industrial design and test process, robotics and many others (Sauer and Schömer, 1998; Baraff and Witkin, 1992; Baraff, 1992; Tasora, 2001; Milenkovic and Schmidl, 2001).

In simulation and Virtual Reality environments, the interaction with the scene is one of the most important parts that have to be taken into account. The physical interaction between the objects must be properly modeled in order to reproduce the system's reaction properly

(Stewart, 2000; Song et al., 2000; de Jalón and Bayo, 1994; Baraff and Witkin, 1992).

Impact Management

The computational simulation of contacts in rigid body systems has to deal with two major aspects: on one hand, the physical model of contacts, and on the other hand, the computational approach to the corresponding model.

The subject of contact modeling has been extensively studied and new models for specific situations are still being developed (Kry and Pai, 2003; Song et al., 2000; James and Pai, 2001). There are some assumptions common to all of them that define the problem (Shabana, 2001; de Jalón and Bayo, 1994).

When two rigid bodies impact against each other, forces act between them preventing inter-penetration. If the surfaces of the bodies in contact are smooth enough, a plane tangent to both surfaces can be defined. This determines a normal vector \mathbf{n} which goes outwards one of the bodies. In this situation, the relative velocity of the contact points is defined as

$$\mathbf{v} = \mathbf{v}_2 - \mathbf{v}_1$$

where \mathbf{v}_1 is the velocity of the contact point on the body from which \mathbf{n} goes outward, and \mathbf{v}_2 is the velocity of the contact point on the other body.

It is generally assumed that an impact is taking place only when the contact points are approaching. If the relative velocity in the normal direction $v_n = \mathbf{v} \cdot \mathbf{n}$ is positive, it means that the points are moving apart. This case is not considered as an impact, as in the immediate next instant of time there will be no collision.

When a contact is produced, two situations can arise; the forces can act exclusively in the direction of the normal vector, or they can also act in the direction tangent to the contact (within the plane defined by the contact). In the second case, the contact is said to be *frictional*.

The approach to the contact problem usually requires an additional classification of the contacts according to the normal relative velocity (Baraff, 1992; Baraff and Witkin, 1992; de Jalón and Bayo, 1994; Stewart, 2000). A contact is said to be a resting contact when the relative normal velocity of the contact points v_n is null. On the other hand, a contact is said to be a non-resting contact when the relative normal velocity v_n is negative.

In the implementation of contacts, a threshold is defined so that, whenever a contact is detected, it is classified as resting or non-resting. Then the corresponding model is used to compute the response of the system.

This paper presents a simplified model for management of contacts to be used in dynamic simulation. Section 2 overviews the classical models for resting and non-resting contact, including some aspects of their computational implementation. In section 3, we justify the validity of the assumptions taken in this work regarding the accuracy of the models. In the following sections, the new model is presented and discussed. Some numerical tests and a comparison of the new model with the classical methods are presented. Finally some remarks are given about more specific questions such as friction models.

CONTACT MODELS

Resting Contact Model. Coulomb Friction

When a resting contact takes place, an equilibrium exists among the external forces that push the two bodies against each other, and the reaction forces of the contact. In this situation the contact forces will never make the contact points to start moving. Therefore, the only forces that the contact produces are those opposite to the normal component of the external forces pushing the bodies together.

Now let's consider a situation with p contact points. The force at every contact acts as an external force when calculating the response for the other contacts. The calculation of all the forces for this case involves the solution of an optimization problem, whose formulation can be found in (Baraff, 1992), and a more detailed analysis of its implementation aspects in (Baraff, 1994).

When a resting contact takes place and the tangential relative velocity of the contact points, $\mathbf{v}_t = \mathbf{v} - v_n \mathbf{n}$, is not null, friction forces can be considered. The model used to obtain these forces is Coulomb's law (Baraff, 1991; Kry and Pai, 2003; Kraus et al., 1997; de Jalón and Bayo, 1994).

According to this law, the tangential force \mathbf{f}_t is always opposite to the relative tangential velocity of the contact points \mathbf{v}_t . Its magnitude will be proportional to the modulus of the normal contact force \mathbf{f}_c ,

$$\mathbf{f}_t = -\mu |\mathbf{f}_c| \frac{\mathbf{v}_t}{|\mathbf{v}_t|}$$

where μ is the *coefficient of sliding friction*. Figure 1 shows the graphical representation of Coulomb's law.

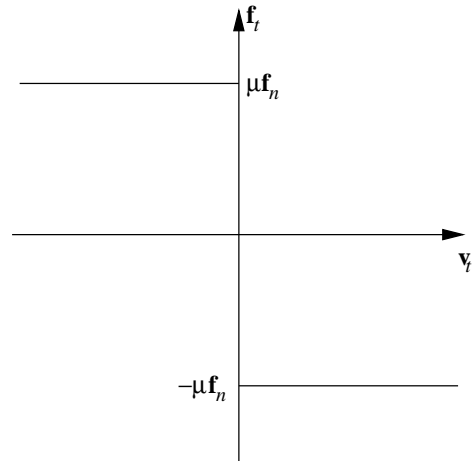


Figure 1: The friction force as a function of the relative tangential velocity

The computational implementation of the problem for p different contact points with friction is more complex, and it has a higher cost, than the frictionless case (Baraff, 1994).

Non-Resting Contact Model

The case of non-resting contact is simpler than the one presented above regarding the calculation of the forces. However, some aspects that we will review in brief make its implementation quite tricky.

The main assumption for the non-resting contacts model is that the force of the contact acts only during an infinitesimal interval of time. This kind of forces are called impulses and they cause discontinuities in the derivatives of the system (Shabana, 2001; Stewart, 2000; de Jalón and Bayo, 1994).

The effect of the impact impulse is that the contact points start moving apart at the same instant of the impact. Their velocity depends on how much energy is lost in the impact. The rate between the module of the normal velocity before and after the contact is indicated by a parameter called *restitution coefficient*, denoted by e . If $e = 1$ the contact is said to be *elastic*, and the normal velocity is strictly inverted. If $e = 0$ the normal velocity is null after the impact, and it is said to be a *plastic contact*.

If the impact takes place at time t_c the normal velocity of the particle after the impact $v_n(t_c)^+$ is obtained by the expression

$$v_n(t_c)^+ = -e v_n(t_c). \quad (1)$$

As an example, let's consider a particle with mass m and position \mathbf{x} and a wall represented by a plane with normal vector \mathbf{n} . In this case the relative velocity of

the contact is the velocity of the particle $\mathbf{v} = \dot{\mathbf{x}}$, and the normal direction will be the vector perpendicular to the wall. Figure 2 shows a diagram representing the velocities before and after the impact in the example.

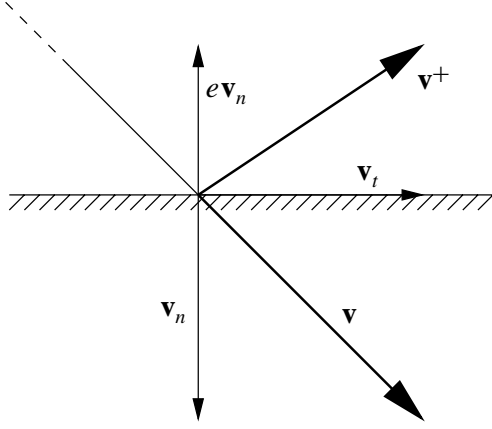


Figure 2: The velocity of a particle after a contact with a wall, with restitution coefficient e

This model corresponds to the frictionless case. The study of non-resting impacts with friction is out of the scope of this paper and won't be discussed here.

The non-resting contact model requires to compute, not only the impulse that acts in the contact, but also the change in the velocity of the bodies.

The impulse due to the impact is obtained from the relative normal velocity of the contact points after the impact determined by equation (1). This value can be expressed analytically in an explicit form (Baraff, 1992). Once the value of the impulse is known, the new values for the velocities of the two bodies (not of the contact points but of the bodies) can be computed.

Furthermore, all these calculations have to be done for the instant of time when the impulse acts. Thus they cannot be done within the normal integration loop as the moment when the contact happens most probably will not belong to the discretization considered for the simulation.

The usual way to tackle this problem is to stop the integration whenever an impact is detected. A root finding method is used to calculate an approximation of the instant when the impact took place, t_c , which is added to the discretization of the simulation process.

Following, the state of the system is stepped backwards to the last step before the contact. Then, the state is integrated until t_c and the impulse is obtained and applied. The integration is resumed after this process (Baraff, 1992; Shabana, 2001).

In order to avoid instabilities produced by the application of the impulse, a reduction of the step size during a short time interval can be necessary (Baraff, 1992).

LOW ACCURACY MODELS

The complexity of the computational aspects of the models for impact management in rigid bodies simulation, makes the dynamic models to be quite complex to implement and to have some additional computational cost.

In order to overcome with the drawbacks induced by the models, we propose a much simpler technique which is based in relaxing the accuracy of the computational results. The simplicity of the new model ensures high performance, and the most important, real-time capabilities.

Of course, this new technique cannot be applied to every simulation process, but we consider that it is good enough to be used in those applications related with 3D virtual environments, like driving or plane simulators.

All these applications are worried about achieving a realistic representation of the world they simulate and the movements of the objects being displayed take a crucial part in this goal. Therefore the resolution of the dynamic models must match, or at least must be of the same order that used for the visual models, which in most cases is not very high at all.

Some works have been published in order to ease the calculation of the impacts to make them suitable for real-time simulation (Bouma and Jr., 1993; Baraff, 1994; Milenkovic and Schmidl, 2001) but none of them has used other method but different variations to the one described in the previous section.

NEW MODEL FOR NON-RESTING CONTACTS

The basis of the proposed method is the assumption that the contact forces can be considered not as instant impulse forces, but as another external force, that acts during an interval of time. Considering this, the main drawbacks shown above due to the impulse nature of contact can be avoided.

Therefore, our main objective will be the calculation of such forces to obtain a system behavior very similar to the one predicted by the theoretical model.

In order to simplify the calculations we will only consider the case of a particle, though later in this paper some details are given about the problem for 3d rigid bodies. We will also consider an scenario where appears no other force than those caused by the impact. As a matter of fact this does not suppose a loss of generality, as in the proposed method the contact forces are applied like external forces.

The formulation will be obtained from the example given in a previous section that presented a particle in contact with a wall.

The main condition that must be fit is the condition imposed by equation (1). This means that the force applied must invert the velocity of the particle in the amount imposed by the *restitution coefficient*. In order

to approximate to the theoretical model, this force will be strong enough to invert the normal velocity in a single step of the integration.

Another condition that we want to be satisfied is that the contact forces on the particle must always oppose to the contact, and never act in favor of it. This will prevent from situations where the objects get glued to each other as a result of the application of the contact forces. Therefore, the force has to be applied until we detect that the normal velocity of the particle goes in the opposite direction to the normal of the wall. Then, the velocity will remain constant until the contact finishes.

Depending on the integration method used in the simulation, such force will have a different expression. We will show how to estimate its value using the Explicit Euler method, but the procedure can be applied to a general case.

According to this method, the velocity at step t_{i+1} is computed from the velocity at step t_i and the forces acting on the particle by:

$$\mathbf{v}(t_{i+1}) = \mathbf{v}(t_i) + h\mathbf{f}/m$$

Assuming that only the contact force \mathbf{f}_c is considered, it will be collinear to the direction determined by \mathbf{n} , so it will only affect to the normal component of the velocity. If the equation has to fit the condition $v_n(t_{i+1}) = -ev_n(t_i)$, then

$$\mathbf{f} = -\frac{m}{h}v_n(t_i)(e + 1)\mathbf{n}$$

Thus, this expression can be applied to calculate the force of the contact in a simple way.

To demonstrate the method some numerical experiments have been done. The position of a particle of mass $m = 1$ has been integrated, using an Explicit Euler method with an integration step of $h = 0.005$. Several simulations have been done for $e = 0.2$, $e = 0.5$ and $e = 0.9$. As starting condition, it has been set a velocity of $\dot{\mathbf{x}}(0) = -1$ and, in order to represent the results clearly, a different starting position has been set for each value of e .

Figure 3 shows the position of the particles versus time. Both the integration result and the solution according to the theoretical model have been represented.

It can be observed that after the impact there is a slight difference between the computed and the theoretical position. As the lines have the same slope, the computed value for the velocity and the value predicted by the theoretical model are the same. To confirm this, an estimation of the restitution coefficient was also obtained dividing the value of the integrated velocity just before and after the impact

$$\hat{e} = -\frac{v_n}{v_n^+},$$

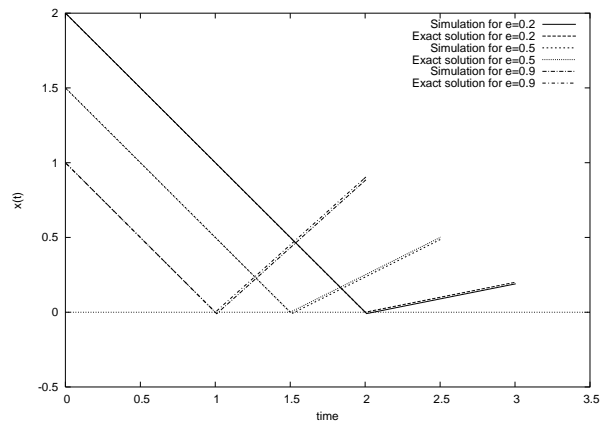


Figure 3: Representation of the theoretical and computed values of the position of the particle versus time for different values of e

and in all cases it has matched the original restitution coefficient.

Therefore, the only source of discrepancy of the new method against the usual one, is caused by the delay that is produced while the particle is inside the wall. The exact difference can be calculated computing the depth of the penetration that occurs until the velocity is inverted. It will depend on the starting condition, but always it will be lower than twice the integration step multiplied by the normal velocity.

NEW MODEL FOR RESTING CONTACTS

The presented scheme for impact management has been defined and tested for the case of non-resting contacts. Now let's see what is the behavior of the new model in the case of resting contacts.

According to the rules of our model, if a particle is left on the surface of the floor, no force will act on it due to the contact. If a constant acceleration such as gravity is applied a velocity will appear at the next integration step. Then, the model will make a reaction force to act on the particle, but the gravity acceleration will not allow that the velocity of the particle is completely inverted. Due to this, a residual normal speed will remain and the particle will start descending at a speed almost constant. This makes the presented model inadequate to manage resting forces.

Figure 4 shows the value of the position and the velocity along time of a particle that is initially resting at $x = 0$. The *restitution coefficient* has been set to $e = 0.5$ and a constant acceleration of -1 is being applied. This is equivalent to a gravitational field of $g = -1m/s^2$. In the figure it can be clearly observed how the rules described before affect to the particle; it starts moving due to the action of the constant acceleration and eventually the velocity becomes stable. Then the particle keeps on penetrating into the wall at almost constant velocity.

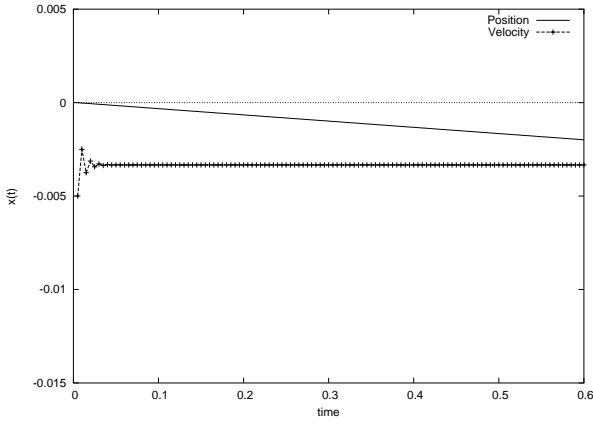


Figure 4: The position and the velocity of a particle resting on a wall under constant acceleration

In consequence, the method has to consider in a different way the force to be applied to the particle, in order to deal with this situation and to achieve the desired behavior.

As in previous section, the method will be presented for simplicity in the case of a particle that is in contact with (in this case, resting on) a wall or the floor. In the case that the contact involves the interaction of 3-dimensional bodies, the model defined for the particle has simply to be applied simultaneously to every contact point that appear and no additional constraint has to be considered, in contrast with the classical approach.

Let's now redefine the model. A pair of forces will be applied in two steps. The first force will be applied as in the previous model, but instead of provoking the inversion of the normal velocity, it will be calculated to make it null after one integration step.

With the particle stopped, a damped spring model will be applied to oppose to the possible forces that impulse the particle deeper into the wall.

$$\mathbf{f}_k = -(kd + cv_n)\mathbf{n}$$

Note that this model will not prevent penetration. The spring constant will determine how strong is the non-penetration constraint. However, if we know a bound of the forces that are expected to appear during the simulation, a value for k can be found that does not allow more than certain violation of the constraint. On the other hand, the election of the damping term c determines how quick the velocity of the particle dies out.

As we expect to observe a similar behavior in particles or bodies with different masses, as in the previous section, the force of the contact is multiplied by the mass of the body.

As a final remark, although the stiffness term of the force always impulses the particle in the direction of the normal vector, the damping term of the force can

impulse the particle against the floor if $\mathbf{v} \cdot \mathbf{n} > 0$. This situation cannot be allowed since, as we said before, the contact forces must never act in the direction opposite to \mathbf{n} . Thus the force that will be applied in practice is

$$\mathbf{f}_k = \begin{cases} -m(kd + cv_n)\mathbf{n}, & \text{if } (kd + cv_n) < 0 \\ 0, & \text{otherwise} \end{cases}$$

This model completes the non-frictional contact management presented here. Whenever a contact is detected, either the resting or the non-resting contact model is used to obtain the forces to be applied. A threshold must have been defined previously to take this decision, as in the classic approach.

The experiment that showed that the first model was not adequate for resting contact has been repeated for the second model. Figure 5 shows the response of both models using the same initial condition: $\mathbf{x}(0) = 0$ and $\dot{\mathbf{x}}(0) = 0$. The parameters have been chosen as $k = 1000$ and $c = 50$. The problems appeared in the previous model have been solved, since the velocity of the particle now tends to zero and its position becomes stable, allowing only a slight penetration.

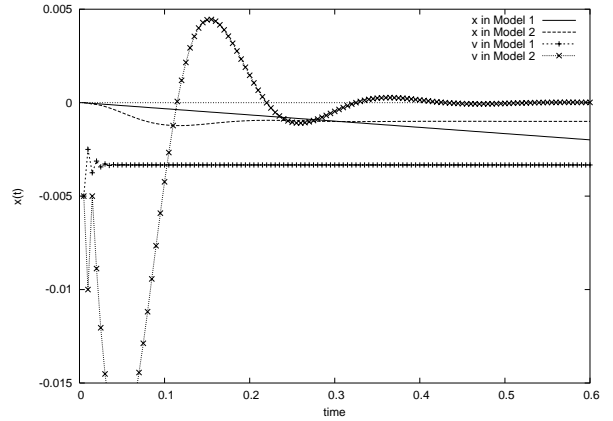


Figure 5: Comparison between the position and velocity of the particle using the first and the second model

Finally, a situation in which both resting and non-resting contacts are involved has been simulated. Figure 6 shows the position versus time of a particle under a gravitational field of $1m/s^2$ with initial condition $\mathbf{x}(0) = 1$ and $\dot{\mathbf{x}}(0) = 0$. It has been chosen a restitution coefficient of $e = 0.5$ for the non-resting impacts. The threshold to change from the first model to the second one has been chosen as $v_n^r = 0.1m/s$. It can be seen how, after several rebounds, the particle rests in the ground.

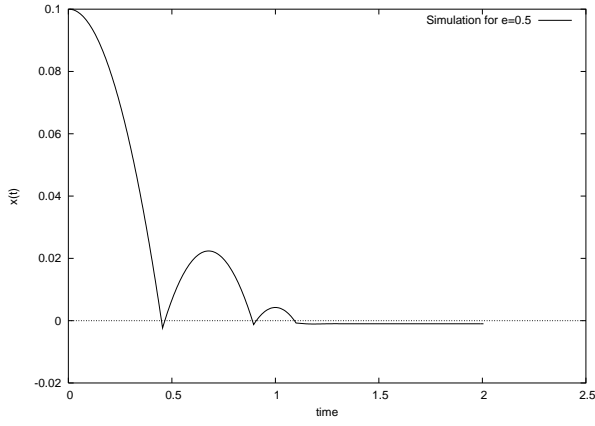


Figure 6: A simulation of a particle falling on the ground

FINAL REMARKS

Frictional Contacts

The models presented have been formulated only for non-frictional contacts. No tangential forces appearing as a consequence of the contact have been considered. However, the use of frictional contacts is quite important, as most models need them to be realistic (Stewart, 2000).

The introduction of the friction model for the resting contact using the presented method is straightforward. Remind that the tangential force that appears in a frictional contact is

$$\mathbf{f}_t = -\frac{\mathbf{v}_t}{|\mathbf{v}_t|} \mu \mathbf{f}_n \quad (2)$$

where \mathbf{v}_t is the relative tangential velocity of the contact points, \mathbf{f}_n is the normal force due to the contact and μ is the *coefficient of sliding friction*. The application of this expression can be done in a very direct manner; all the variables in equation (2) are known, so it is enough to introduce them in the expression and apply the resulting force.

Note that the damping coefficient of the resting model can influence in the behavior of the sliding force. If it is very low, the oscillations of the normal force due to the variation of the normal velocity can cause the tangential force to act as a series of impulses. A smooth solution will be achieved using the adequate damping factor.

Application of the results

The work presented in this paper has been applied successfully in the development of several harbor crane simulators. The project involves the development of a complex simulation system for five harbor cranes for the purpose of training operators at the Valencia Harbor. The proposed method has substituted the classical approach in the impact management module of the dynamical models.

Figure 7 shows two screen captures of contacts during a real time simulation.

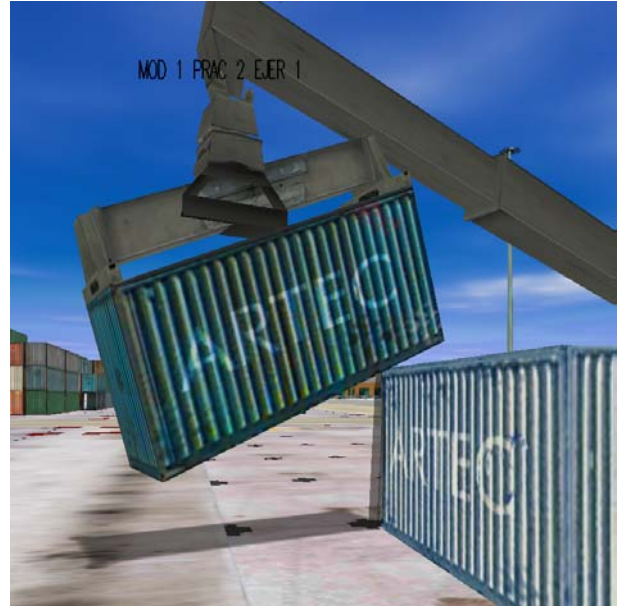
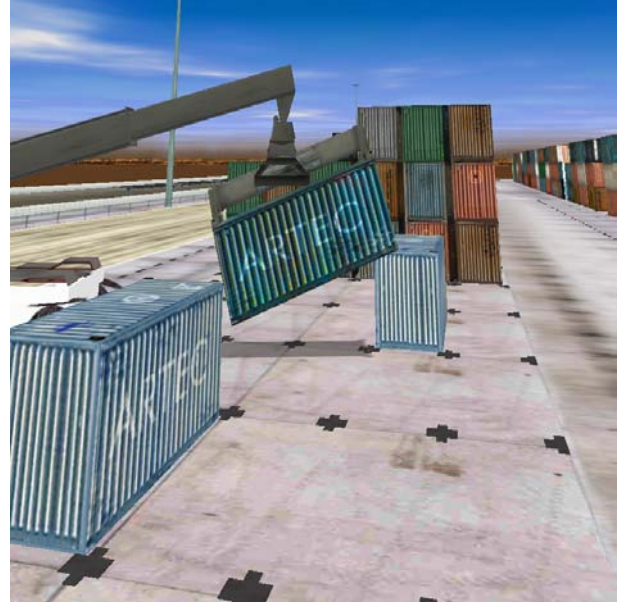


Figure 7: Two screen images of contacts during the simulation of the Reach-Stacker crane simulator at Valencia Harbor

CONCLUSIONS AND FUTURE WORK

We present a new contact model and its associated contact management approach. It overcomes the main difficulties that arise in the implementation of contact models, due to the consideration of impacts as instantaneous impulses and the strict consideration of the non-penetration constraint.

The result is a method that is completely embedded in the integration loop. No backward stepping nor step size reduction is needed to get the contact response. For the case of punctual masses, the solution obtained with this method has the same velocity as the solutions obtained with the traditional procedure. It only presents some differences in the position, due to the fact that it allows a slight violation of the non-penetration constraint. This difference will be of the same order than the step size of the integration method within a wide range of situations.

The calculation of the forces is also easier than in the usual approach, using a simple spring and damper scheme. The expressions of the forces are calculated previously according to the integration method used in the simulation.

In future works, we will perform more exhaustive numerical tests to obtain a precise estimation of the range of validity of the model, mainly in the case of 3-dimensional objects. However, the successful application of the method in a complex simulation system, such as the mentioned crane simulator, indicates that the results will be good. An improvement for the 3-dimensional case can be done by the substitution of the expressions for the forces proposed for the case of non-resting contact by others obtained from the impulse expression of the theoretical model. This would increase the accuracy of the solution, in spite of using a more complex expression for the force.

REFERENCES

- Baraff, D. (1991). Coping with friction for non-penetrating rigid body simulation. In *Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, pages 31–41. ACM Press.
- Baraff, D. (1992). Dynamic simulation of non-penetrating rigid bodies. Ph. D Thesis 92-1275, Computer Science Department, Cornell University.
- Baraff, D. (1994). Fast contact force computation for nonpenetrating rigid bodies. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 23–34. ACM Press.
- Baraff, D. and Witkin, A. (1992). Dynamic simulation of non-penetrating flexible bodies. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 303–308. ACM Press.
- Bouma, W. J. and Jr., G. V. (1993). Modeling contacts in a physically based simulation. In *Proceedings on the second ACM symposium on Solid modeling and applications*, pages 409–418. ACM Press.
- de Jalón, J. G. and Bayo, E. (1994). *Kinematic Kinematic and dynamic simulation of multibody Systems: The Real-Time Challenge*. Springer-Verlag.
- James, D. and Pai, D. K. (2001). A unified treatment of elastostatic and rigid contact simulation for real time haptics. *Haptics-e, the Electronic Journal of Haptics Research*, 2(1).
- Kraus, P., Fredriksson, A., and Kumar, V. (1997). Modeling of frictional contacts for dynamic simulation. In *Proceedings of IROS 1997 Workshop on Dynamic Simulation: Methods and Applications*.
- Kry, P. G. and Pai, D. K. (2003). Continuous contact simulation for smooth surfaces. *ACM Transactions on Graphics (TOG)*, 22(1):106–129.
- Milenkovic, V. J. and Schmidl, H. (2001). Optimization-based animation. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 37–46. ACM Press.
- Sauer, J. and Schömer, E. (1998). A constraint-based approach to rigid body dynamics for virtual reality applications. In *Proceedings of the ACM symposium on Virtual reality software and technology 1998*, pages 153–162. ACM Press.
- Shabana, A. A. (2001). *Computational Dynamics*. John Wiley & Sons, 2 edition.
- Song, P., Kraus, P., Kumar, V., and Dupont, P. (2000). Analysis of rigid body dynamic models for simulation of systems with frictional contacts. Technical report, University of Pennsylvania - Department of Computer and Information Science, <http://www.cis.upenn.edu/techreports>.
- Stewart, D. E. (2000). Rigid-body dynamics with friction and impact. *SIAM Review*, 42(1):3–39.
- Tasora, A. (2001). An optimized lagrangian multiplier approach for interactive multibody simulation in kinematic and dynamical digital prototyping. In *VII ISCSB*, Milano.

SIMULATION IN AEROSPACE AND TRAFFIC

Aerospace Simulation

BRIDGES IN HLA DISTRIBUTED SIMULATIONS

Benoît Bréholée
Pierre Siron
ONERA-CERT
Information Modelling and Processing Department
2, av. E. Belin
BP 4025
F-31055 Toulouse Cedex 4
France
E-mail : {breholee|siron}@cert.fr

KEYWORDS

HLA, distributed simulation, domains, inter-federations, bridges.

ABSTRACT

In this paper we describe our current works concerning HLA inter-federations. Inter-federations are the connection of several HLA simulations. We present the interests of such simulation architectures. Then we describe the problems raised by the implementation of the bridges needed to build these simulations. We indicate the solutions we considered in the case of our bridge prototype. Then we present our current results (implementation of inter-federations and performance tests) and on-going works.

HLA DISTRIBUTED SIMULATIONS

HLA is a standard for distributed simulations, initially proposed by the US Department of Defence, then accepted as an IEEE open standard (IEEE 1516). Reuse and interoperability are fundamental goals of HLA. This led to the proposed topology for HLA simulations (Figure 1): complex simulations (called federations) are realized by interconnecting simpler simulators (called federates). These federates actually connect to a simulation kernel called RTI (runtime infrastructure). Federates manage common objects and exchange information as defined in the HLA specifications.

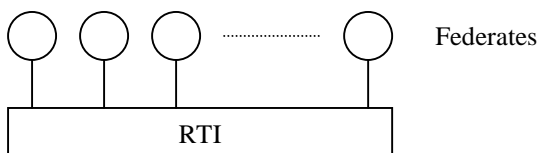


Figure 1 : Federation architecture

HLA Specifications

The HLA specifications consist of a set of rules about federates and federations, an object model for the federates (and particularly for public objects and interactions appearing in the federations), and a set of services required to manage objects and interactions. Some of these services have to be provided by the federates to the RTI, others have

to be provided by the RTI to the federates. Particularly, federates publish and subscribe to relevant object and interaction classes, so as to be able to create such entities or to receive information about them.

HLA at ONERA

Several works have been carried out at ONERA (Office National d'Études et de Recherches Aéronautiques) in the distributed simulation field, and particularly with HLA. One of our main results is the development of a RTI prototype called CERTI (Bréholée and Siron 2002). Other projects include airports of the future (Adelantado 1999), security extensions to the RTI to manage the inter-company problem (Bieber et al. 1998), and simulations of radar detection of missiles and drones designed by the Departments of Physics of ONERA.

We are mainly motivated by the scalability and the security problems. CERTI being under development (not all HLA services and language bindings are implemented), interoperability with other RTI is another significant issue.

INTER-FEDERATIONS

Notion of Domain

Several on-going works concern the introduction of the concept of domain in CERTI and/or in federation executions:

- The architecture of our RTI prototype is currently based on one central process (RTIG) and local processes accompanying every federate (RTIA). In large-scale simulations the RTIG is a bottleneck and a distributed RTIG may improve this situation.
- The HLA provides optimisation services which are not currently implemented. Publications and subscriptions can be refined with *Data Distribution Management* services to deal with attribute value ranges, instead of global classes.
- Inter-federations (federations interconnections) realized with bridge federates may be used to create topologies similar to the distributed RTIG one, but are useful for interoperability and security purposes too.

Inter-federations

Inter-federations are sets of interconnected federations. The aspects of inter-federations we are particularly interested in are:

- **Interoperability:** a bridged federation can use several RTI. We are interested in this solution to cope with federates using different language bindings and RTI missing adapted language binding support. A Java federate may be used in an inter-federation involving our (currently C++ only) RTI prototype, with an intermediate RTI.
- **Security:** with inter-federations, it's possible for different parties to manage a significant part of the simulation, including one or more RTI and bridges. In an inter-federation, RTI may use different Federation Object Models (FOM), and events appearing in one federation may remain local to the federation if the connected federations are not able to manage them or don't recognize them.
- **Optimisation:** inter-federations provide new topologies for HLA simulations, in particular without the unique central process (RTIG in the case of CERTI) that appeared to be the bottleneck in large-scale simulations. First, we are interested in the implementation of inter-federations involving the federates of an overloaded federation. This experiment provides the case of a similar simulation, with a reduced number of federates per RTI (though the number of objects and events is the same). Then, other interesting experiments consist in defining as much local objects and interactions as possible, to introduce domains and reduce irrelevant data exchanges. Hierarchical federations (Cai et al. 2001) are based on this principle to create federate subsystems actually made of federations.

The connection of federations is realized with bridge federates, at the application level. Any other link between federations would bring significant problems: inter-RTI communications, not defined in the HLA specifications, would be limited to single RTI vendor experiments.

HLA BRIDGES

An HLA bridge is actually a set of interconnected federates, each federate belonging to one particular federation. A typical implementation is a single process software joining two federations as a federate, and creating objects in both federations.

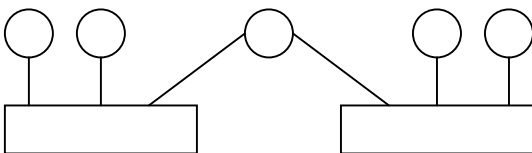


Figure 2 : Inter-federation bridge

In simplest inter-federations, the goal of a bridge is to observe and simulate federations: each federation is observed by a bridge federate, and simulated into other federations. For example, any object created in a “source” federation is seen by a bridge federate, then a proxy object is created by the bridge in a “target” federation. The bridge aims at controlling the proxy object so that its behaviour is identical to the original object: when an attribute value changes, the corresponding attribute value in a proxy object should be changed accordingly. Particularly when time is involved, events have to occur at the same logical time.

General Translation Problem

Since a bridge operates at the application (federate) level, there is no control on the elements' references in the federations: object and interaction classes are referenced with a handle, which is different from one federation to another. This is true for dynamic elements too: handles of objects, interactions, etc. This is why the bridge has to track and translate all the references involved in the joined federations. This leads to the architecture proposed in (Braudaway and Little 1997): an inter-federation bridge, joining 2 federations, is based on 2 federates and a Transformation Manager. This component is necessary to translate references corresponding to the same entity. But we also consider translation for values, in the case of differences in scales used by the federates (including time scales).

Main Update Mechanisms

The main mechanisms concern direct changes in the data shared by the federates. Examples of such information are the creation and deletion of object instances, attribute values updates, sending interactions, etc. These mechanisms consist of two service calls: a federate request and an update message sent by the RTI to all the relevant federates. For the subscribed federates, the update is received as only one unconditional message: there is no other message related to the actual change. This is why this mechanism can be directly translated by a bridge federate in another federation (Figure 3). As soon as an object creation is detected, a proxy object can be registered. Attribute value updates of this object can be reflected to the proxy object's attributes.

Time Management

In coordinated time simulations, previous messages can carry timestamps to indicate the event's logical time. Such timestamps have to be translated between federations. In HLA federations, federates have to be declared *constrained* when they may receive time-stamped messages, and *regulator* when they send such messages. This is why an inter-federation bridge has to be both constrained and regulator in the connected federations. If not, messages may lose their *time-stamp order* status through the bridge. Being constrained and regulator, the bridge has to regularly advance in time, and to provide a lookahead. This

lookahead can be determined with only source federations information, since the bridge does not create non-proxy objects or interactions. The actual lookahead associated with the current logical time in a target federation corresponds to the lower bound on time stamps (LBTS) of incoming messages. In the case of a two-federation inter-federation, this value is directly provided by a HLA service. If the bridge connects more than two federations, this lower bound is the minimal value of the source federations lower bound on time stamps.

This is the method we have chosen to handle time management over inter-federations, since this works in the case of our federations. But this is not enough for more general inter-federations, particularly with zero lookahead situations: this problem is raised in (Lake 1998), where extensions to provide inter-federation level LBTS are proposed as a solution.

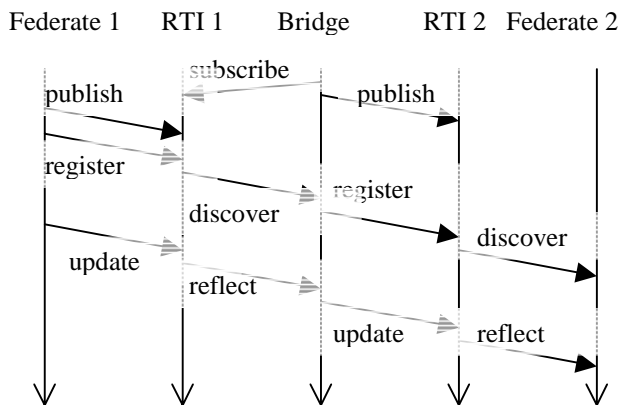


Figure 3 : Translation of direct mechanisms

Consensus Mechanisms

Some mechanisms such as federation save and restore processes, and synchronization cannot be directly translated by a bridge with the usual HLA services.

The general behaviour of a bridge is to represent one or more federations. In the case of objects and interactions, it provides proxy objects and proxy interactions whose behaviour is the same as the original one. But some aspects to be translated correspond to the state of the federations globally. For example, a “federation saved” status represents a whole federation. The bridge is expected to declare itself saved when the federation it represents is successfully saved (and if it represents several federations, when they are all successfully saved). But since every connected federation is both a source and a target, such a behaviour would lead to a deadlock during the save process (or any other consensus-based mechanism), as detailed in (Braudaway and Little, 1998 ; Dingel et al. 2002). In a save process, the final save result is sent by the RTI, and depends on the results of each federate’s save process. Particularly, it is required that all federates succeed in their save process. In one federation, after the save process is initiated, the RTI waits for these results. The bridge, as a

federate, has to provide a save process status. But this status depends on the federation it represents. If the save mechanism has been translated, the RTI in this federation is waiting for federates’ save process results, including the bridge. There is a deadlock situation since the bridge will not provide a response to any federation until one of them is actually saved.

One solution available from the HLA specifications is the use of the *Management Object Model*: federates can use this structure to receive information they usually shouldn’t need, such as RTI services used by other federates. The save process case handled with the MOM is detailed in (Braudaway and Little 1997): the bridge federate has to observe the federates’ save results sent to each RTI to conclude on the inter-federation global save result. The MOM can be used in such cases, when the bridge lacks critical information. But it is a solution we did not consider mainly because it is not implemented in CERTI yet, so the solution wouldn’t be usable in the cases we are interested in. We would have to study the consequences of this federation-level information available to the federates before using it too.

We considered several solutions for the consensus mechanism problem. The most obvious case concerns the services not currently implemented in CERTI: they are simply ignored (As is described in (Dingel et al. 2001), ignoring the translation of some mechanisms may be an acceptable solution, if the simulations to implement allow it). This is the case of the save and restore process, which we don’t need in our federations. But, if such mechanisms were required, the solution we consider consists in restricting the use of these services: the deadlock in these mechanisms disappears as soon as the bridge provides a result to a RTI. In some federations, the federate that is initially requesting the save process is the only federate to actually use the save result. Then, the representation of this federation’s status, in other federations, is not needed. So our solution consists in allowing the bridge to indicate a success although the represented federation is not saved. If this is done for all federations except the one where the request was done, then the bridge can provide a correct result to this federation, and particularly to the requesting federate. The restriction consists in forbidding the use of the save result for other federates, since it may reflect only a federation save process, and not the whole inter-federation save process.

This method would not work with synchronization, another consensus-based mechanism: all the federates have to receive a result and to be able to trust this result. Synchronization is mainly used in our federations during the federation initialisation. These cases can be handled with particular bridge configurations, but these are not generic solutions. The only general solution is the use of the MOM services.

Data Filtering

Some scenarios do not require the representation of objects and interactions with proxies to concern all the information usually shared by the federates. In some security related situations, parts of the federations have to remain inside those federations and should not be transferred through the bridge. The same requirement exists for optimisation purposes, to avoid irrelevant subscriptions and data transfers. Finally another category of situations where data filtering is required concerns cyclic inter-federations: these inter-federations involve several federations and bridges, and allow several “paths” from one federation to another. But to keep consistency, the bridges involved have to represent only some parts of the federations, to avoid any redundant proxy, which would actually lead to an infinite number of proxies (Damon et al. 1998). This is why we have considered the implementation of data filtering in our bridge prototype. It is currently only based on object names (for test purposes) but many criteria may be considered.

TOPOLOGIES

With the proposed solutions to the translation of HLA mechanisms through a bridge federate, several inter-federation topologies are possible. Without data filtering, any acyclic inter-federation can be realized (provided that the necessary HLA mechanisms are translated). In this case, all the bridge instances are identical and do not need any particular data-related configuration: they translate any event from “source” federations to “target” federations (each federation assuming both roles, depending on the events translated). Figure 4 is an example of acyclic inter-federation, based on several 2-federate federations.

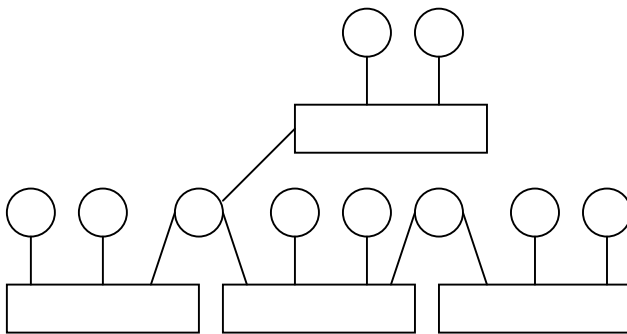


Figure 4 : Acyclic inter-federation

Data filtering brings the ability to implement cyclic inter-federations. Mechanisms can be translated and events can be represented with the acyclic inter-federation methods. The only condition concerns the inter-federation consistency: events to be represented should have one and only one associated proxy event. Figure 5 is an example of cyclic inter-federation.

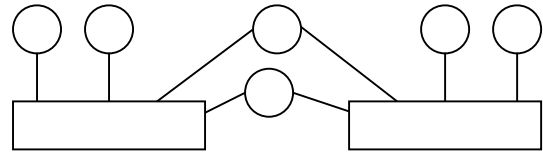


Figure 5 : Cyclic inter-federation

RESULTS AND ON-GOING WORKS

Our first result is the realization of inter-federations thanks to the implementation of a bridge prototype. We focused on the design of inter-federations using CERTI, our RTI prototype, in at least one involved federation. The set of HLA mechanisms this bridge is currently able to translate from one federation to another directly depends on the available services of CERTI, and on our needs in our federations.

So the current implementation of this prototype focuses on direct mechanisms such as creating and destroying objects, updating attribute values, sending interactions. As detailed in a previous section, consensus-based mechanisms are not translated for now, and we are considering a solution based on restrictions. Coordinated time simulations are possible with the translation of time stamped messages but zero lookahead is not supported. Both acyclic and cyclic inter-federations were realized. Data filtering of “local” objects for optimisation purposes was implemented too.

Though this first step focused on the realization of the bridge, not optimization, we conducted several tests to compare the performance of federations and their associated inter-federations in various topologies. Most inter-federation topologies are slower than the associated single-federation simulation: bridges represent additional transfers and processing. The only inter-federation providing better results than the associated single federation was based on one bridge interconnecting many CERTI-based federations: we compared an 8-federate federation and the associated inter-federation based on 8 federations (each one supporting only one of the previous federates). With the same duration, the time coordinated federation advanced of 840 time steps, while the inter-federation advanced of 1191 time steps. Other kinds of inter-federations were slower, but we do not try to interpret these results now since further works concerning optimisation should be done first.

On-going works concerning the bridge focus on two aspects: implementation of missing translations and optimisations. No optimisation effort was done in this first prototype, but possible improvements concern the detection of irrelevant subscriptions, and configuration of routing regions: *Data Distribution Management* services of HLA,

focused on optimisation, could be used to refine the filtering operated in the bridge.

CONCLUSION

In this paper, we described the principles and interests of HLA inter-federations. We detailed several problems in the realization of inter-federation bridges, and the solutions we considered for a first prototype. We presented possible topologies for these inter-federations. Our main result is the implementation of several kinds of inter-federations, with different topologies. We carried out performance tests, and noticed one topology improving the speed of time coordinated simulations using CERTI, but more performance tests should be realized after further optimisation works.

On-going works concern these optimisations, either based on irrelevant data exchanges filtering or on bridge configuration and inter-federation design. Further works concern new translations: other mechanisms, values (for example for federates using different time scales). Finally, another goal is the implementation of inter-federations involving different RTI (particularly CERTI and the DMSO RTI-NG) for interoperability purposes.

REFERENCES

- Adelantado M. 1999. "Experimenting the HLA framework for the ONERA Project Airport of the Future". In *Proceedings of the 1999 Fall Simulation Interoperability Workshop*.
- Bieber P., J. Cazin, P. Siron and G. Zanon. 1998. "Security Extensions to ONERA HLA RTI Prototype". In *Proceedings of the 1998 Fall Simulation Interoperability Workshop*.
- Braudaway W. and R. Little. 1997. "The High Level Architecture's Bridge Federate". In *Proceedings of the 1997 Fall Simulation Interoperability Workshop*.
- Bréholée B. and P. Siron. 2002. "CERTI: Evolutions of the ONERA RTI Prototype". In *Proceedings of the 2002 Fall Simulation Interoperability Workshop*.
- Cai W., S. J. Turner and B. P. Gan. 2001. "Hierarchical Federations: An Architecture for Information Hiding." In *Proceedings of the 15th Workshop on Parallel and Distributed Simulation*.
- Damon C., R. Melton, R. Allen, E. Bigelow, J. Ivers and D. Garlan. 1998. "Formalizing a specification for analysis: the HLA ownership properties". Technical report CMU-CS-98-149, Department of Computer Science, Carnegie Mellon University.
- Dingel J., D. Garlan and C. Damon. 2002. "Bridging the HLA: Problems and Solutions". In *Sixth IEEE International Workshop on Distributed Simulation and Real Time Applications*.
- Lake T. 1998. "Time Management over Inter-Federation Bridges". In *Proceedings of the 1998 Fall Simulation Interoperability Workshop*.

AUTHORS BIOGRAPHIES

BENOÎT BRÉHOLÉE was graduate from a French engineering school of aeronautics and computer science in 1999. Then he is a PhD student at ONERA. He is a member of the Design and Validation of Computer Systems research unit.

email: breholee@cert.fr
web: <http://breholee.free.fr>

PIERRE SIRON was graduate from a French engineering school of computer science in 1980, and received his doctorate in 1984. Then he is a Research Engineer at ONERA and he works in parallel and distributed systems and computer security. He is a member of the Design and Validation of Computer Systems research unit.
email: siron@cert.fr
web: <http://news.cert.fr/francais/deri/siron/>

GENETIC ALGORITHM FOR SCHEDULING AIRCRAFT LANDING

Mohamed BACCOUCHE, Jaouad BOUKACHOUR, Abdellatif BENABDELHAFID

ISEL LABORATOIRE CERENE

Quai frissard BP 1137

76063 Le Havre, FRANCE

E-mail: baccouchemohamed@voila.fr, boukachour@iut.univ-lehavre.fr, benabdelhafid@iut.univ-lehavre.fr

KEYWORDS

Landing, Programming, Optimization, Scheduling, Genetic Algorithm, Air Traffic Control.

ABSTRACT

The problem of scheduling aircraft (plane) landings is one of deciding a landing time on a runway for each plane within a predetermined time window (target time) and separation criteria between the landing of plane, and the landing of all successive plans, are respected. There are penalties associated with landing either earlier or later than this target landing time.

In this paper, we consider the static single runway aircraft landing. We also present an effective genetic algorithm for the problem. The computational results are presented for a number test problems involving up to 10 planes and one runway.

INTRODUCTION

The appellation Evolutionary Algorithms (EAs) is a generic term to describe the computer approaches based on the principles evolution as key elements of their functioning. There is a large number of evolutionary algorithms: genetic algorithms, evolutionary programming, evolutionary strategies and genetic programming[Koza92]. All these approaches have in common the simulation individual structures evolution through methods of selection, mutation and reproduction. The genetic programming is a genetic algorithm extension to programs. That means that individuals are programs expressed in the shape of syntactic trees.

We are, in a first time, going to formulate the landings planes problems, then we will propose a synthesis, a resolution approach and finally present results.

THE GENETIC ALGORITHMS

FORMATTING THE MANUSCRIPT

1. Basis concept

The genetic algorithms are exploration algorithms based on the natural selection mechanism and on genetics. They are optimization processes of that are inspired by the biologic behaviors of the natural populations, and are based on the evolution of "large number" of solutions to

the treated problem, the evolution being guided by one fitness functions that is maximized during the process[Cotta95]. We will present the further on concepts, proprieties and heuristic of adaptation of the GA along with a general idea of the main works as for as planes landings are concerned.

In genetics a set of parameters represented by a particular chromosome is called its genotype, the phenotype is a particular process of a genotype. The assessment function depends on the phenotype. The transposition of these biologic concepts in an artificial setting leads on the following concepts definitions:

- Population: It is the set of individuals, or the set of chromosomes of a same generation.
- Function fitness: This function is determined according to the problem to solve and the coding chosen for chromosomes. To a particular chromosome it assigns a numeric value, that is supposed proportional to the individual interest as a solution. In the case of a function optimization, the fitness is in a trivial way the function that one tries to optimize, but it is not always the case.

2. Genetic operators

The moving from a population to another is made by the selection. Then the selected individuals undergo the mutation then the recombination.

- Selection: According to the individuals quality, each one has a percentage to be chosen for reproduction, witch corresponds to the relative importance of the individual quality in relation to the population total quality.
- Crossover: It consist in the exchange of genes between two chromosomes to create new chromosomes.
- Mutation: The mutation, starts by the selection of one of the chromosome followed by the change of its value with a predefined probability, called mutation probability.

PROBLEM DESCRIPTION

The schedule of planes landings requires several treatments spaced by time in relation to the imposed lengths, with a advance and delay margin. It is therefore a

landing organization of for which planes are linked by a separation constraint of in an imposed time interval. we must respect to best timetables of takeoff and landing of planes, otherwise the airport undergoes a " penalty of advance " by minute of advance or a " penalty of delay "(by minute of delay).

Physically the plane can always arrive in airport between the earliest date E_i and the latest date L_i (E_i if it flies to maximal speed, and L_i if it turns in circle until weariness of the critical fuel). Data concerning flights can be regrouped in this table: [Andreas 2001, Beasley 1998].

Let:

N the number of planes

E_i be the earliest landing time for plane i ($i=1, \dots, N$)

L_i be the latest landing time for plane i ($i=1, \dots, N$)

T_i be the target "preferred" landing time for plane i ($i=1, \dots, N$) where $E_i \leq T_i \leq L_i$

S_{ij} be the required separation time of (≥ 0) between plane i landing and plane j (where plane i lands before plane j), $i=1, \dots, N$; $j=1, \dots, N$; $i \neq j$

Av_i be the penalty cost (≥ 0) per unit of time for landing *before* the target time T_i for plane i ($i=1, \dots, N$)

Ret_i be the penalty cost (≥ 0) per unit of time for landing *after* the target time T_i for plane i ($i=1, \dots, N$)

Then variables of decision are:

X_i the landing time for plane i ($i=1, \dots, N$)

$b_{ij} = 1$ if plane i lands before plane j ($i=1, \dots, N$; $j=1, \dots, N$; $i \neq j$)
 $= 0$ otherwise

α_i how soon plane i ($i=1, \dots, N$) lands before T_i

β_i how soon plane i ($i=1, \dots, N$) lands after

The objective function consist to:

$$\text{Minimize } \sum_{i=1}^N (\alpha_i Av_i + \beta_i Ret_i)$$

Under the constraints:

$$\begin{cases} E_i \leq X_i \leq L_i & i = 1, \dots, N \\ b_{ij} \in [0, 1] & i, j = 1, \dots, N; i \neq j \\ b_{ij} + b_{ji} = 1 & i, j = 1, \dots, N; j > i \\ X_j \geq X_i + S_{ij} - b_{ji} * (L_i - E_j + S_{ij}) & i, j = 1, \dots, N; i \neq j \end{cases}$$

The problem discussed in this paper concerns the sequencing and scheduling of N arrival flights to one runway. The problem is formulated with the following conditions:

- For each of the N arrival flights, an estimated time of arrival E is available as the minimum time to get to the runway
 $E = \{129, 195, 89, 96, 110, 120, 124, 126, 135, 160\}$;
- When the flight is scheduled to a runway, the delay as a result of the schedule is defined as the difference between the scheduled time of arrival TA and the earliest time E .
- Any landing sequence must satisfy the requirements for separation time, which are specified in terms of separation matrix:

$$S_{ij} = \begin{bmatrix} 0 & 3 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 \\ 3 & 0 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 \\ 15 & 15 & 0 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 15 & 15 & 8 & 0 & 8 & 8 & 8 & 8 & 8 & 8 \\ 15 & 15 & 8 & 8 & 0 & 8 & 8 & 8 & 8 & 8 \\ 15 & 15 & 8 & 8 & 8 & 0 & 8 & 8 & 8 & 8 \\ 15 & 15 & 8 & 8 & 8 & 8 & 0 & 8 & 8 & 8 \\ 15 & 15 & 8 & 8 & 8 & 8 & 8 & 0 & 8 & 8 \\ 15 & 15 & 8 & 8 & 8 & 8 & 8 & 8 & 0 & 8 \\ 15 & 15 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 0 \end{bmatrix}$$

3. The genetic algorithm elements

- Fitness function:

The techniques of scale and penalty apply, by defining an appropriate fitness function is delicate. The major difference is to know if a program is good or not, or if we must execute it one or several times with different data in entries or in different contexts etc. In our case, the fitness function is written:

$$\text{Fitness}(s) = \text{Maxof}(P) - \text{Minimize } \sum_{i=1}^N (\alpha_i Av_i + \beta_i Ret_i)$$

where s is a solution of population P and $Maxof$ is a function which return the maximum value of the objective function of the solutions in the population P .

- Genetic operators:

Three operators are used: selection operator chooses the more adapted individuals (those having the high score), matching operator (cross-over) and finally mutation operator (mutation).

- Encoding:

We use a real encoding. A chromosome contains the scheduling times of N arrival flights. The following chromosome S describe the arrival times of N flights where $S[i] = TA[i]$ such each $TA[i]$ belongs to $[E_i, L_i]$.

262	406	98	115	127	218	139	147	163	180
-----	-----	----	-----	-----	-----	-----	-----	-----	-----

- Selection:

The selection mechanism is based on the roulette well selection [Golberg89].

- Crossover:

We propose a crossover based on an exchange by blocks of elements between two chromosomes to generate two others. There are several ways to proceed this exchange according to the constraints of the treated problem. A point k is randomly selected to separate a parent into two sections. The first section is directly copied into child. The remaining genes of the child are orderly fulfilled from another parent. Each resulting children, contains genes inherited from each parent. The crossing principle is defined as follows:

Evaluate initial population:

Father 1	262	406	98	115	127	218	139	147	163	180
Father 2	290	262	110	102	123	151	138	163	197	209

Create New chromosomes:

Child 1	262	406	98	102	123	151	138	163	197	209
Child 2	290	262	110	115	127	218	139	147	163	180

We uses the following algorithm:

```

Begin
int i, k;
k=1 + (int)((float)7*rand()/(RAND_MAX));
If (i<=k) Then
Child1[i]=Father1[i];
Child2[i]=Father2[i];
i=i+1;
Else
Test of the chromosome genes

```

```

Child1[i]= Father2[i];
Child2[i]= Father1[i];
i=i+1;
Endif
End

```

- Mutation:

The mutation is a shape of the individual genetic adaptation to his environment. it works by a brutal modification of a gene.

The mutation principle is defined as follows:

Evaluate new chromosomes:

Child1	262	406	98	102	123	151	138	163	197	209
Child2	290	262	110	115	127	218	139	147	163	180

Create new generation:

Father1'	262	406	90	102	123	151	138	163	197	209
Father2'	290	262	110	115	100	218	139	147	163	180

We uses the following algorithm:

```

Begin
Int i, N;
i=0;
for i < N Do
IF (Random(x) < Pmut) Then
S[i]=Genalea (E[i], L[i], S, i);
i=i+1;
Endif
End

```

Where $Random(x)$ returns unreal between 0 and $x-1$, $Pmut$ is the rate of mutation and $Genalea(e, l, s, i)$ return a time between e and l . This time must satisfy the separation constraints time between others elements of the solution S except the i th element.

The mutation is applied to certain individuals immediately after the crossing. If the random number is lower than the rate of mutation, the chromosome will be muted.

The mutation introduce diversity in the population.

- Algorithm:

```

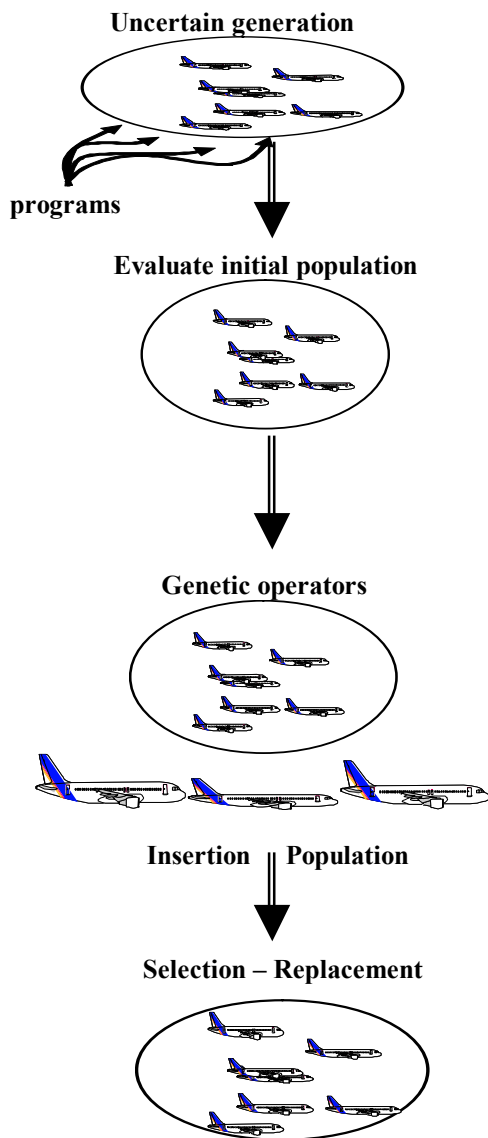
Begin
Generation=0;
Parent Population = Initial population;
While Generation < GenerationMax Do
Selection;
Crossover
Mutation
Test of the chromosome genes
New population

```

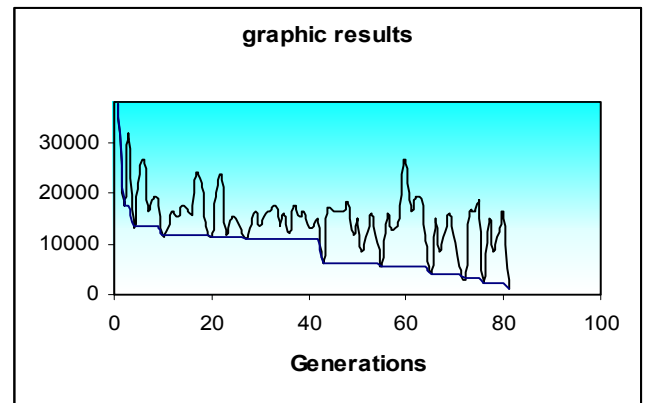
Generation = Generation + 1
End

This process of simulated biological selection-creation-destruction is repeated for a certain number of generations. The best member of this population then represents the answer.

Random chromosomes are selected and evaluated to determine the suitability with respect to the conditions that a result should satisfy.



- Graphic results:



— Evolution of solutions
— Better solution

Conclusion

In this paper, we have presented the problem of scheduling planes landing. We create genetic algorithm that permits to converge toward this goal. On the other hand, we also expand this algorithm in order to treat some specific application fields. But these works, in order not to remain only theoretical and conceptual, also require a practical validation, and this through the construction of operational tools (distributed and a collaborative system, graphic representation of genetic evolution).

The algorithm can be ameliorated by using the binary encoding because we have seen after several generations the population becomes uniform and the crossing doesn't create any more diversity.

If the size of the population is less, a less crossing rate must be combined with an elevated mutation rate. The interest of the mutation is to be able to explore different regions of the research space, guaranteeing therefore that all the space have a non hopeless probability to be tried.

REFERENCES

- [Jaouad 2002] J. Boukachour, A. El hilali Alaoui “ A genetic algorithm to solve a problem of scheduling plane landing” in Proc. of the Int. Conf Advanced Computer System, ACS'02, October 2002, Miedzyzdroje, Poland
- [Andreas 2001] . Andreas T Ernst and Mohan Krishnamoorthy, "Algorithms for scheduling Aircraft landing", CSIRO Mathematical and Information Sciences. Private Bag 10, Clayton South MDC, Clayton VIC 3169, Australia, August 15, 2001.

[Beasley 1998] J.E. Beasley, M. Krishnamoorthy, Y.M. Sharaiha, D. Abramson, "scheduling Aircraft landing-the static case", The management School, Imperial College, London SW72AZ, England, February, 1998.

[Cotta95] C. Cotta, J.F. Aldana, A.J. Nebro, J.M. Troya, "Hybridizing Genetic Algorithms with Branch and Bound Techniques For the Resolution of The TSP", in Proc. of the Int. Conf. on Artif. N.Net. and GAs, April 1995, in Ales, France, pp. 277-280.

[Davis91] L. Davis, The Genetic Algorithm Handbook, Ed. New York: Van Nostrand Reinhold, ch. 17, 1991.

[Goldberg89] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Jan.1989.

[Koza92] J.R. Koza, Genetic Programming : On the Programming of Computers by means of Natural Selection, Cambridge, MA, MIT Press, 1992.

SIMPLIFYING SENSITIVITY ANALYSIS IN SUBWAY CONTROL

Felisa J. Vázquez-Abad
DIRO, Université de Montréal,
C.P. 6128 Succ. Centre-Ville, H3C 3J7 Canada,
email: vazquez@iro.umontreal.ca

Lourdes Zubieta
Bishop's University
Lennoxville, QC, J1M 1Z7 Canada
email: lzubieta@ubishops.ca

KEYWORDS

subway control, sensitivity analysis, score function.

ABSTRACT

We consider a subway network where each platform is served by only one line with headway μ_ℓ . Travel times between platforms are given by a fixed amount plus an additional stochastic noise; the probability distribution of the inter-departure times at any platform is then a convolution of previous iid random perturbations. The performance function is the sum of an operating cost and the passengers' "social" cost of total waiting time. We establish normal approximations to inter departure times when the stochastic noise has bounded variance and use a score function (SF) estimator for the partial derivatives. The estimation of the gradient of the performance function can be decoupled by platform so that we only need to estimate derivatives at transfer stations. Numerical experiments are performed to compare SF with an estimate by finite differences.

INTRODUCTION

Trains in a subway network are frequent enough that synchronization is not an issue; transfer passengers that miss a connection do not have to wait as much as in the case of intercity networks. However, short distances between stations, fluctuations in dwelling time (times that doors remain open for passengers to un board and board), and even conductor's driving patterns cause enough stochastic noise to make inter departure times random. Each line ℓ has an average headway μ_ℓ associated with it representing the average time between adjacent train departures. These headways are the control parameters that Transit Commissions have to determine in order to satisfy demand at reasonable costs.

Consider a subway network where each station has several platforms, one per direction of all lines passing through the station. Single stations serve only one line and so have two platforms while *transfer* stations have $m/2$, ($m \geq 2$) lines and m platforms. To simplify the

notation we consider that each platform in a station is served by only one "line".

The overall performance of the subway system is defined by the sum of two components: a deterministic operational cost, linear in the number of trips per line and proportional to headways, and a "social cost" represented by the waiting time of all passengers multiplied by a constant that represents the cost of one minute of waiting

At a single platform, passengers arrive only from the outside so there is one source of passenger arrivals. At transfer platforms, there are many sources of arrivals: from the outside and from the connecting platforms.

In this paper we consider a fixed time interval, representing a fraction of a day, with passengers arriving at a *constant rate*; for example, arrivals may follow a Poisson process with 2 passengers per minute on average. The number of arrivals is random but the *parameter* of the probability distribution remains fixed. This simplification is for clarity purposes only and the model is easily generalized for many time intervals, each having a different set of arrival rates.

The headway of a line clearly affects the waiting times of passengers starting their travel at platforms along the line, but also affects the waiting times of transferring passengers: those coming *from* connecting platforms and those transferring *to* other platforms in the same subway station. Hence, the global waiting time of transfer passengers depends on all the headways of the lines used by passengers and it can be very difficult to evaluate the sensitivities (derivatives) with respect to each of the headways, an important step for example, to seek the headway values that minimize the performance function. It was shown before (Vázquez-Abad and Zubieta 2000) that this difficulty can be overcome by translating the estimation of the gradient into several ersatz estimation subproblems, one per source of transfer arrivals at each transfer platform. However, we had no numerical evidence of the worth of such ersatz estimators because we

had no efficient way of calculating at least one of the partial derivatives involved in the ersatz calculation..

In this paper we use a score function (SF) estimator (Rubinstein and Shapiro 1993) developed for the Normal inter departure times to estimate some partial derivatives (Vázquez-Abad and Heidergott 2003).. We simulate a transfer platform and test the robustness of this estimator when the inter departure times *are not* normally distributed. The theoretical values of the partial derivatives being unknown, we compare these approximations with estimates obtained by finite differences .

The contribution of this paper is twofold: to test numerically the robustness of the SF estimator to departures from normality and to verify the former theoretical results concerning the ersatz derivatives.

The paper is organized as follows. Next section briefly describes the general model, then we show how to decouple the problem into local models per platform, and estimate partial derivatives with a score function SF. Next we present the ersatz estimation of the gradients, and finally we give computational results comparing the SF estimators to gradient estimates obtained by finite differences.

THE MODEL

We consider here a model proposed before (Vázquez-Abad and Zubieta 2000) A subway network consists of \mathcal{L} lines denoted by $L_1, L_2, \dots, L_{\mathcal{L}}$. A line is described by a sequence of platforms $p_1 \rightarrow p_2 \rightarrow p_3 \dots$ where $p_k \rightarrow p_{k+1}$ denotes that platform p_k immediately precedes platform p_{k+1} . For each platform p there is a unique line ℓ such that $p \in L_{\ell}$.

To ease notation, we assume that there is a unique sequence of platforms a passenger follows for each origin-destination (o,d) pair and this sequence may include several transfers of line. This restriction can be easily removed. If passengers can take several paths to go from their origin to their destination, the model can extend to consider that a passenger may choose a path with a given probability within his/her optimal strategy according to a transit assignment model (Spiess and Florian 1989), or by determining least travel time routes in stochastic, time varying networks (Miller-Hooks and Mahmassani, 2003).

In what follows, we assume that during the period of time considered and for every origin-destination pair (o,d), passengers with destination d arrive at the origin platform o according to a stationary Poisson process with constant rate $\lambda_{o,d}$. All Poisson processes are

independent and independent of the train departure processes.

The Normal Approximation

The inter departure time at any platform, in particular at a transfer platform, is defined as a constant plus a sum of random perturbations. Therefore its probability distribution depends on the probability distribution of such perturbations and by the Central Limit Theorem can be approximated by the Normal Probability Law.

Let $\mu = (\mu_{\ell}, \ell = 1, \dots, \mathcal{L})$ be the vector of headways (control parameters). For each platform p let $\{\delta_j(p)\}$ be a sequence of zero-mean iid random perturbations representing the fluctuations due to passengers boarding/un boarding the j -th train at platform p . For each line ℓ with headway μ_{ℓ} , the train departure epochs V_j at the initial platform $p_1 \in L_{\ell}$ follow the recursion:

$$V_j(p_1) = V_{j-1}(p_1) + \mu_{\ell}(1 + \delta_j(p_1))$$

for all train arrivals within the time period under consideration. Then the inter departure time of train j at the first platform p_1 is defined as

$$\begin{aligned} Y_j(p_1) &= V_j(p_1) - V_{j-1}(p_1) \\ &= \mu_{\ell}(1 + \delta_j(p_1)), \quad j = 1, 2, \dots \end{aligned}$$

Now, if we consider the next platform, say p_2 , the departure epoch of train j (now at p_1) will be equal to the departure epoch at the first platform p_1 plus the travel time from p_1 to p_2 plus the perturbation at platform p_2 :

$$V_j(p_2) = V_j(p_1) + \frac{D(p_1, p_2)}{v} + \mu_{\ell}\delta_j(p_2)$$

where $D(p_1, p_2)$ is the distance between the platforms p_1 and p_2 and v is the average speed of trains. Similarly, the departure epoch for next train $j+1$ at platform p_2 can be expressed as:

$$V_{j+1}(p_2) = V_{j+1}(p_1) + \frac{D(p_1, p_2)}{v} + \mu_{\ell}\delta_{j+1}(p_2)$$

and so the inter departure time of train $j+1$ at platform p_2 is given by:

$$\begin{aligned} Y_{j+1}(p_2) &= V_{j+1}(p_2) - V_j(p_2) \\ &= \mu_{\ell}(1 + \delta_{j+1}(p_1) + \delta_{j+1}(p_2) - \delta_j(p_2)) \end{aligned}$$

Similarly, at platform p_3 :

$$\begin{aligned} Y_{j+1}(p_3) &= \mu_{\ell}\{(1 + \delta_{j+1}(p_1) \\ &\quad + \delta_{j+1}(p_2) - \delta_j(p_2) + \delta_{j+1}(p_3) - \delta_j(p_3))\}. \end{aligned}$$

In general, the inter departure times at any platform p_k are described by a sum of the differences of random

perturbations and can be written as follows:

$$Y_{j+1}(p_k) = \mu_\ell \{ (1 + \delta_{j+1}(p_1) + \sum_{h=2}^{k-1} (\delta_{j+1}(p_h) - \delta_j(p_h))) \} \quad (1)$$

The train departure processes are dependent along the platforms of the same line as indicated by equation (1) but independent from any other departure process, given the assumption of no synchronization.

From equation (1) it is clear now that the inter departure times Y_j are convolutions of random variables. By the Central Limit theorem, we know that the probability distribution of $Y(p_k)$ tends *approximately* to the Normal distribution, regardless of the probability distribution of the δ_j 's and as long as these are iid and have bounded variance. In our case, the number of terms in (1) may be small when a transfer platform is close to the starting platform and it is difficult to tell in advance what is the minimum number of terms required to obtain a good approximation in terms of average transfer waiting times and of the estimation of derivatives. In the last section of this paper we show results for several distributions of δ_j . In what follows, it is assumed that the μ_ℓ 's are *scale parameters* of the distribution of inter departure times, and that the initial time V_0 is independent of μ .

Transfer Waiting Times

Consider a transfer platform p . Let \mathcal{M} be the number of connecting platforms in the same station as p and call these p_m , ($m = 1, \dots, \mathcal{M}$). Let S_k^m represent the arrival epochs of train k from connecting line m having P_k^m passengers in transfer to platform p , for $k = 1, \dots, N_m(V_j)$, where $N_m(V_j)$ is the number of such transfer arrivals in the time period considered. The total waiting time for transfer passengers coming from a connecting platform m and boarding train j at platform p can be written as:

$$W_{p,j}^m(\mu) = \sum_{k=1}^{N_m(V_j)} P_k^m (V_j - S_k^m).$$

It has been proved (Vázquez-Abad and Zubieta 2000) that if the passenger arrivals are Poisson, the number of transfer passengers arriving by an incoming train k , P_k^m , is conditionally Poisson distributed with conditional mean $\lambda_p^m T_k$, where λ_p^m is the aggregated rates of all O/D pairs requiring a transfer from platform p_m to p . Then the total transfer waiting time for transfer passengers m for all trains departing from platform p during the time interval of length T can be approximated by

$$W_p^m(\mu) \approx \sum_{j=1}^{M(T)} \lambda_p^m \sum_{k=1}^{N_m(V_j)} T_k^m (V_j - S_k^m). \quad (2)$$

where $M(T)$ indicates the (random) number of trains departing from platform p during the time interval T . Notice that (2) is not a continuous function as an infinitesimal perturbation in the parameter μ may shift an arrival epoch V_j so that a group of transfer passengers can board train j in the original process and have to wait for train $j + 1$ in the perturbed process. Finally, the performance function we consider here is the sum of all transfer waiting times at all transfer platforms:

$$W(\mu) \approx \sum_{\ell=1}^{\mathcal{L}} \sum_{p \in L_\ell} \sum_{m=1}^{\mathcal{M}} W_p^m(\mu) \quad (3)$$

GRADIENT ESTIMATION

One goal of this paper is to study how the normal approximation described in the previous section affects the estimation of the sensitivities of the total transfer waiting time. To estimate the gradient of (3) we must remember that the headway of a line clearly affects the waiting times of passengers starting their travel at platforms along the line, but also affects the waiting times of transferring passengers: those coming *from* connecting platforms and those transferring *to* other platforms in the same subway station.

To obtain sub-optimal values of the control parameters μ it is necessary to follow a sequential algorithm that requires unbiased estimators of the derivative of $W_p(\mu)$ with respect to each μ_ℓ at every transfer platform p :

$$\frac{dW_p(\mu)}{d\mu_\ell} = \sum_{m \in \mathcal{I}(p)} \frac{\delta E[W_p^m(\mu)]}{\delta \mu_\ell} + \sum_{n \in \mathcal{O}(p)} \frac{\delta E[W_p^n(\mu)]}{\delta \mu_\ell} \quad (4)$$

where $W_p^m(\mu)$ represents the waiting time of passengers transferring from platform m to platform p , $W_p^n(\mu)$ denotes the waiting time of passengers transferring from p to platform n , $\mathcal{I}(p)$ is the set of connecting platforms with passengers transferring to p and $\mathcal{O}(p)$ denotes the set of connecting platforms to which passengers from p can transfer to.

If the inter departure times (1) are i.i.d. random variables distributed according to a Normal distribution with mean μ and standard deviation $\mu\sigma$, we can develop unbiased estimators for the first term of (4) using the score function method (Vázquez-Abad and Heidegott 2003) as follows:

$$\frac{\delta E[W_p^m(\mu)]}{\delta \mu_\ell} \approx E[L_p^m(\mu)] = E[S_p(\mu_\ell) \cdot W_p^m(\mu)], \quad (5)$$

where

$$S_p(\mu_\ell) = \frac{1}{\mu_\ell^3 \sigma_\ell^2} \sum_{j=1}^{M(T)} (Y_j^2(p) - \mu_\ell Y_j(p)) - \frac{M(T)}{\mu_\ell}.$$

where the term $S_p(\mu_\ell)$ multiplying the total transfer waiting time is the derivative of the Normal probability distribution. This approach can be extended to determine an estimate for the partial $\delta E[W_p^m(\mu)]/\delta\mu_m$. Let $N(T)$ represent the number of transferring train arrivals from platform m during the time period T . The derivative of the expected waiting time of passengers coming from platform m with respect to μ_m is:

$$\frac{\delta E[W_p^m(\mu)]}{\delta\mu_m} \approx E[L_p^m(\mu)] = E[S_m(\mu_m) \cdot W_p^m(\mu)], \quad (6)$$

where

$$S_m(\mu_m) = \frac{1}{\mu_m^3 \sigma_m^2} \sum_{k=1}^{N(T)} (T_k^m(p)^2 - \mu_m T_k^m(p)) - \frac{N(T)}{\mu_m}.$$

The derivative (6) can be calculated by platform p and sent to the connecting platform m to complete the estimation of its partial derivative. Notice that this approach requires several local calculations at every transfer platform: one with respect to its line plus one for each connecting platform. Once this is done, they must send (6) to each platform m , receive the estimates for the second term of (4) and finalize its own partial derivative estimation. The number of required estimates is therefore proportional to the number of platforms in the network.

Ersatz Partial Derivatives

An alternative to estimating directly equation (6) was proposed before (Vázquez-Abad and Zubietta, 1999) for departure and arrival stationary processes. Using a scaling argument, the derivative of W with respect to μ_m can be written in terms of the derivative with respect to μ_ℓ , reducing even more the computational effort.

Denote by E^* the conditional expectation given that the time elapsed since the last arrival $S_0^m = 0$. If $E^*[W]$ is continuously differentiable in μ_ℓ and μ_m ,

$$\frac{\delta E^*[W_m^p]}{\delta\mu_m} = 2 \cdot \frac{E^*[W]}{\mu_\ell} - \frac{\mu_m}{\mu_\ell} \cdot \frac{\delta E^*[W_m^p]}{\delta\mu_\ell} \quad (7)$$

The next section contains a comparison between the two estimators: (6) which requires extra calculations during the simulation run and (7) that can be evaluated once the two estimates in the right hand side of (7) are obtained.

SIMULATION RESULTS

The first experiments were aimed at testing whether normal approximations give accurate results in terms of the total transfer waiting time. Given that the inter

departure times in (1) are convolutions of iid random perturbations $\delta_j(p)$, the distribution of $Y_j(p)$ at transfer stations may be approximated by random normal variables $\mathcal{N}(\mu, \theta\mu)$, where μ is the headway of the line passing by the transfer platform p where passengers are waiting. (Chung 1979).

The tests were done with two different probability distributions: Uniform and Weibull. We measured the transfer waiting time at a platform q of a line 1 for passengers coming from the third platform $q = 3$ of line 2. The headways used were $\mu_1 = 4.0$, $\mu_2 = 7.3$ and the simulations run for 100,000 periods of 60 minutes each.

The density function used for the Uniform random perturbations was $U(-0.1, 0.1)$ for the first platform $q = 1$ of a line and $U(0, 0.02)$ for all other platforms of the line. This means that the inter departure times which are generated by a difference $\delta_{j+1}(p_q) - \delta_j(p_q)$ are triangular. The density function used for the Weibull distribution was: $f(x) = \alpha \beta^{-\alpha} x^{\alpha-1} e^{-(x/\beta)^\alpha}$ for $\alpha = 2$ (arbitrarily) and $\beta = 0.017117$. The parameters of the Normal distribution used were $\mu = \mu_1$ and the variance was set. If the perturbations are Uniforms then the variance of the convolution of (1) at platform q is: $\sigma^2 = \mu_1^2 \cdot \delta^2/3.0 + \gamma^2(q-1)/12$. This value of σ was used to generate the normal random variables and to obtain the values for the parameters b and β so that all distributions being compared had the same variance.

We observed that the statistics of the average waiting times at transfer platforms do not differ significantly when using the actual convolution and when using its normal approximation as shown in Tables (1) and (2). This is important because this quantity will be required for the estimation of the partial derivatives later on. The confidence intervals shown correspond to a significance level $\alpha = 0.05$

Table 1. Normal Approximation with Uniform δ'_j s

q	Avg W(Normal)	Avg W (Uniform)
1	108.83 \pm 0.13	108.91 \pm 0.13
2	108.88 \pm 0.13	108.97 \pm 0.13
3	108.83 \pm 0.13	109.04 \pm 0.13
4	108.90 \pm 0.13	109.10 \pm 0.13
5	109.10 \pm 0.13	109.12 \pm 0.13
6	109.08 \pm 0.13	109.32 \pm 0.13
7	109.13 \pm 0.13	109.24 \pm 0.13
8	109.18 \pm 0.13	109.32 \pm 0.13
9	109.20 \pm 0.14	109.46 \pm 0.14
10	109.22 \pm 0.13	109.45 \pm 0.14

Table 2. Normal Approximation with Weibull δ'_j s

q	Avg W (Normal)	Avg W (Weibull)
1	108.83 \pm 0.13	108.84 \pm 0.13
2	108.94 \pm 0.13	108.91 \pm 0.13
3	109.00 \pm 0.13	108.99 \pm 0.13
4	109.01 \pm 0.13	108.94 \pm 0.13
5	108.97 \pm 0.13	109.00 \pm 0.13
6	109.22 \pm 0.13	109.02 \pm 0.13
7	109.08 \pm 0.13	108.99 \pm 0.13
8	109.14 \pm 0.13	109.16 \pm 0.13
9	109.19 \pm 0.13	109.13 \pm 0.13
10	109.29 \pm 0.14	109.31 \pm 0.13

We then proceeded to test the score function estimators (5) and (6) that assume normal inter departure times when these are not normally distributed. Again we perform experiments with the Uniform and the Weibull distributions. To generate the Uniform perturbations we used: $\delta_j(p_q) \sim U(-\delta, \delta)$ for all q with $\delta = 0.1$. The values of the parameters for the Weibull distributions were $\alpha = 2$ and $\beta = 2.15865522\delta\sqrt{1.0/3.0}$ to have the same variance than the Uniform. Headways used in the simulations were set to $\mu_1 = 2.1$, $\mu_2 = 1.9$, $\mu_3 = 1.5$.

We calculated two estimates of the derivatives of the waiting time of passengers in transfer. The first one uses the score function as defined by (5) and the second estimate is obtained by finite differences:

$$\frac{F(\mu) - F(\mu - \Delta)}{\Delta}$$

with $\Delta = .01$ as a benchmark. The Score Function results were obtained by simulating 100,000 time periods of $T = 60$ minutes each and those by finite differences required 10^6 time periods of the same length. For the experiments in tables (3) and (4) we assumed that the inter arrival times from the transferring platform follow a Uniform distribution with $\delta_q \sim U(-0.1, 0.1)$ for all q and that the transferring platform was the third one ($q = 3$) on line 2.

Results are shown in Table (3) for Uniform perturbations. We can see that when perturbations are uniformly distributed, the SF estimates are accurate even for small values of q , that is, for platforms near the beginning of the line. The CPU times are indicated in parenthesis below each 95% confidence interval. Notice that the estimates by finite differences required much higher CPU times. The confidence intervals are larger

for transfer platforms near to the beginning of the line as expected because for small values of q normal approximations should not be very good.

Table 3. Comparison with Uniform departures

q	$\delta E[W_p^m(\mu)]/\delta\mu_1$ by	
	Score Function	Finite Differences
2	24.21 \pm 13.32 (11 sec.)	32.03 \pm 2.13 (161 sec.)
3	31.49 \pm 9.48 (13 sec.)	32.70 \pm 2.09 (216 sec.)
4	32.15 \pm 7.85 (15 sec.)	30.98 \pm 2.11 (247 sec.)
5	32.08 \pm 6.88 (13 sec.)	29.79 \pm 2.12 (287 sec.)
6	30.66 \pm 6.21 (14sec.)	30.83 \pm 2.12 (327 sec.)
7	27.36 \pm 5.75 (16 sec.)	30.45 \pm 2.12 (368 sec.)
8	31.44 \pm 5.38 (17sec.)	30.04 \pm 2.12 (484 sec.)
9	28.65 \pm 5.12 (18sec.)	30.60 \pm 2.12 (438 sec.)
10	32.99 \pm 4.87 (20 sec.)	31.98 \pm 2.12 (477 sec.)

Table 4. Comparison with Weibull departures

q	$\delta E[W_p^m(\mu)]/\delta\mu_1$ by	
	Score Function	Finite Differences
2	38.49 \pm 13.35 (16 sec.)	33.35 \pm 2.13 (211 sec.)
3	26.968 \pm 9.49 (22 sec.)	31.74 \pm 2.09 (411 sec.)
4	28.15 \pm 7.86 (51 sec.)	32.37 \pm 2.14 (554sec.)
5	30.83 \pm 6.89 (44 sec.)	30.49 \pm 2.18 (737 sec.)
6	29.29 \pm 6.24 (43 sec.)	31.08 \pm 2.19 (815 sec.)
7	29.96 \pm 5.75 (49 sec.)	29.09 \pm 2.21 (954 sec.)
8	26.67 \pm 5.37 (57 sec.)	29.42 \pm 2.21 (1089 sec.)
9	31.43 \pm 5.11 (64 sec.)	30.65 \pm 2.22 (1222 sec.)
10	30.51 \pm 4.89 (72 sec.)	32.34 \pm 2.23 (1399 sec.)

We now consider inter departure times that follow a Weibull distribution and the results of the estimates appear in Table (4). The accuracy of the estimates is poor

for low values of q as seen by both the average values and the width of the 95% confidence intervals. However, we notice that the accuracy improves as q increases; this is shown by the smaller width of the confidence intervals for larger values of q . Notice also that the Score Function estimates take less than 10% of the time required to obtain an estimate by Finite Differences.

Finally, table (5) shows a comparison of the accuracy of the ersatz derivative and the SF estimator (6) for different values of q_2 's and q_1 's when the inter departure and inter arrival times at platform q_1 and q_2 respectively are Uniform with averages $\mu_1 = 1.9$ and $\mu_2 = 2.1$. In table (6) the perturbations follow a Weibull distribution with the same parameters as before. We can see in both tables that the ersatz derivatives are better for transfer platforms away from the first one.

Table 5. Comparison with Uniform Perturbations

q_1	q_2	$\delta E[W_p^m(\mu)]/\delta\mu_2$ by	
		SF Derivative	Ersatz Derivative
2	4	27.14 ± 6.71	33.30
3	4	27.25 ± 6.76	26.94
6	4	27.93 ± 6.89	28.35
6	2	25.48 ± 11.80	28.24
6	7	26.50 ± 4.94	26.57
6	10	27.63 ± 4.08	26.34

Table 6. Comparison with Weibull Perturbations

q_1	q_2	$\delta E[W_p^m(\mu)]/\delta\mu_2$ by	
		SF Derivative	Ersatz Derivative
2	4	25.74 ± 6.74	20.45
4	4	28.71 ± 6.78	30.14
6	4	28.55 ± 6.83	29.55
2	9	25.08 ± 4.21	28.33
4	9	27.88 ± 4.26	26.92
8	9	29.27 ± 4.36	30.16

CONCLUSION

We tested a Score Function estimator of the partial derivatives of total transfer waiting times that assume Normal inter departure times at transfer platforms. The distribution of such times is in fact a convolution of the sum of differences of perturbations that, according to the Central Limit Theorem will approximately follow a Normal distribution as the number of terms in the sum increases. We showed that the total transfer waiting time is robust if inter departure times follow a Normal, a Uniform, or a Weibull distribution as long as their variances are equal and this even for low number of terms in the convolution.

The Score function estimates with respect to the platform line have the benefit of requiring only a fraction of time of that required by the method of Finite Differences. The accuracy of the estimates is quite good when the true perturbations follow a Uniform Distribution but they may be poor for transfer stations close to the start of the line if the perturbations follow another distribution, for example a Weibull. For these cases it is possible to develop a different SF estimator based on the empirical distribution of the perturbations. However, for most subway systems of large cities, the SF estimates tested here should be robust to departures from normality. Finally, the ersatz estimates of the derivative with respect to the arriving line rated very well for Uniform and Weibull distributions when the transfer platform is away from the beginning of the subway line. More tests are being currently done with a complete subway network where the stochastic approximation will use the gradient estimators here presented.

REFERENCES

- Chung, K.L.1979. *Elementary probability theory with stochastic processes*, 3rd edition. Springer-Verlag, NY.
- Ross, S.M. 1993. *Introduction to Probability Models*, 5th edition. Academic Press, San Diego.
- Rubinstein, R. and A. Shapiro. 1993. *Discrete Event Systems: Sensitivity Analysis and Optimization by the Score Function Method*. Wiley.
- Spiess, H. and M. Florian. 1989. "Optimal strategies: a new assignment model for transit networks". *Transportation Research B23*, 83–102.
- Vázquez-Abad, F. and B. Heidergott. 2003 "Sensitivity analysis for a problem in public transportation: A comparison of SPA, Score Function and MVD", manuscript.
- Miller-Hooks, E. and H. Mahmassani. 2003. "Path comparisons for a priori and time-adaptive decisions in stochastic, time varying networks". *European Journal of Operations Research 146(1)*, 67–82.
- Vázquez-Abad, F. and L. Zubieta. 2000. "Distributed stochastic approximations for adaptive frequency allocation in subway networks". In *Proceeding of the 39th CDC Conference*. IEEE Society, 4729–4734.
- Vázquez-Abad, F. and L. Zubieta. 2003. "Ghost Simulation Model for the Optimization of an Urban Subway System", Technical report No. XXX GERAD. Université de Montréal.

MORE PUNCTUALITY IN REGIONAL TRAFFIC – USE OF COMPUTER AIDED RESCHEDULING WITH SOUTH GERMAN REGIONAL RAILWAY.

Wilfried Koch

Ravensburg-Weingarten University of Applied Sciences
POB 1261

88241 Weingarten, Germany
E-mail: koch@fh-weingarten.de

KEYWORDS

Scheduling, Transportation train control, transport control, traffic optimization, intelligent transport

ABSTRACT

Optimal rescheduling in cases of delay is important for improving punctuality and service quality in public transportation systems. Since it is impossible to solve the rescheduling task manually, computer aided systems are necessary for this purpose. In the past 10 years the Ravensburg-Weingarten University of Applied Science has conducted research activities in the field of computer aided rescheduling. The results have now reached the stage where the Hohenzollerische Landesbahn AG in Hechingen, Baden-Württemberg, Germany is the first regional railway company in Germany to use computer aided rescheduling. This paper deals with the project as well as with the program system used. It shows some practical experiences and outlines possibilities for future use.

MOTIVATION

Public transportation needs to increase its attractiveness in order to compete with individual means of transportation. Fundamentals of this attractiveness are among others high level of safety, reasonable fares, and reliability.

From the customer's point of view, reliability means a **guaranteed time of travel** with disregard to if transfers are necessary or not.

The work described in this paper covers the reliability aspect of public transportation. Delays extend the travel time not only of the train primarily affected, but also of connecting trains and on single track lines they even affect trains running in the opposite direction than the originally delayed train. Delays prevent traffic from operating according to a published timetable. Instead, the traffic will operate on a corrected schedule which today in most cases is far from optimal and is known to the dispatcher only.

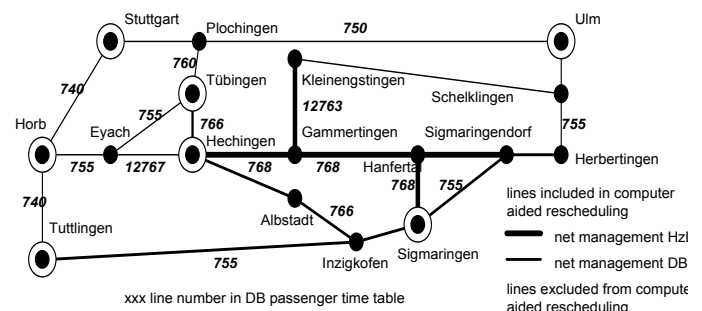
Even for systems and track layouts that at the first glance seem simple, relocating the points of crossing and forecasting the best-corrected timetable is similar to playing chess. Only the use of a computer simulation can guarantee the necessary quality, reproducibility, and speed of the timetable correction. **This is a true but often overseen fact!**

Due to the author's experience as a long distance commuter, he has noticed the large need for rescheduling aids for all sorts of public transportation systems. His personal experi-

ences have made him focus on a low cost solution for regional rail transportation.

Regional public transportation with its many single-track lines has a large need for rescheduling tools and will profit greatly from their use. Such innovations are extremely necessary if regional public transport is intended to become a real alternative to individual traffic. However, in most cases regional public transportation is not the place for big investments, so a strict concentration on a low-cost solution would hopefully ease the program's introduction into practice.

Figure 1: Pilot Area and Surroundings



THE PROJECT

As the preliminary work on the topic showed promising results, we were able to start a project together with a regional railway company in 2001. The goal of this project is to implement, test, and evaluate the rescheduling aid at the Hohenzollerische Landesbahn AG in Southern Germany. The network of our pilot project is shown in Figure 1. The project area is limited by the stations at Herbertingen, Tuttlingen, and Tübingen. The line management for the HzL owned lines occurs at Gammertingen, in the middle of the traffic area, whereas the dispatching decisions for the DB owned lines are made in Karlsruhe, more than 100 km away from the lines included in this project.

Since December 2002, the rescheduling program – now called **RegiDisp** – is operational at the HzL's dispatching centre in Gammertingen. The HzL have trains running on all lines covered by the project. They use the program to optimize the relocation of train crossings and to ensure that transfers are met. Another important use is the short-term construction of conflict free schedules for special train services.

Some characteristic parameters of the project:

- total line length covered ca. 250 km,

- There is an average frequency about 1.5 passenger trains per hour and direction, more in peak time.
- Low frequency of goods trains superimposed.
- All single track,
- 4 nodes of the Baden-Württemberg fixed interval time table (3-Löwen-Takt, 3 lions tact) are included,
- Two rail-to-rail transfer stations inside and three at the borders of the disposition area,
- All lines (besides Hechingen HzL Eyach and Gammertingen – Kleinengstingen) are equipped with electric block.

THE SOLUTION

Works on this subject started in 1992. In three steps we developed the prototype of a rescheduling system for regional public transport which is now available. It bases on the following ideas:

- Realistic, cost conscious solution on a ordinary PC with Windows operating system.
- Designed for a limited area of traffic.
- Sufficient response time.
- Complete system consisting of
 - time-table editor,
 - system for dispatcher's dialog,
 - inference machine and
 - result presentation.

The system was designed to solve all types of conflicts:

Occupational conflicts: two trains compete for the same piece of track, running either in the same or in the opposite direction.

Transfer conflicts: transfers for passenger traffic or rolling stock turn around are endangered.

Timetable conflicts: train is simply delayed without causing the above mentioned conflict types.

Data Model

When we started to design our program system only few literature [e.g. Tomii, N. ; Ikeda, H. (1995)] was available. The data model was developed pragmatically to serve the need of regional and branch railways. It allows for modeling practically all topological situations and all timetable data that we need on regional railways. For example, we can define the track topology or important signaling conditions as well as actual and minimum running times. If needed the minimum running time might be calculated dynamically. Keeping the data in a database guarantees a maximum of consistency and integrity of the data.

Starting from a local optimization strategy, we later implemented a globally optimizing algorithm. It is based on an A*-algorithm but uses sophisticated heuristics in the estimation process to reduce computational time. The algorithm

tries to find the *best* conflict free corrected timetable. It not only optimizes inside of one line but also when delayed connected trains transfer an distortion from one line to another.

The best-corrected timetable is the one producing a minimum cost for the whole observed system. Actually there exist no standards for quality measurement. So, we defined a cost function which was easy to be implemented on the one hand and was plausible and acceptable on the other. Analogous to cost functions in control theory, we chose a weighted sum of delays at all stations. Lost connections are expressed in terms of time and added to the sum mentioned before. The weight factors are train and station specific. The basic formula for the cost is

$$C = \sum_{i=1}^m (W_{T_i} \cdot (\sum_{j=1}^n W_{S_j} \cdot \Delta t_{ij})) + \sum_{k=1}^o W_{A_k}$$

W_A : Weight (importance) of connections / transfers.

W_S : Weight (importance) of stations.

W_T : Weight (importance) of trains.

t_{ij} : Delay of train i at station j.

Each node of the A*-tree represents a timetable. Root is the actual timetable with the primary delays inserted, containing conflicts in most cases. The solutions (solution nodes) are timetables without conflicts.

Going from one node to another in the A*- tree at least one actual conflict is solved. In intermediate nodes (nodes not representing a solution) the A*-Algorithm uses cost values which are the sum of a measured and an estimated value.

In order to speed up the search and to reduce storage needs this estimation took a lot of efforts.

Programming Technique

The first versions of our rescheduling aid were hybrid software systems, using Delphi, the Interbase SQL Server and PDC Prolog. These implementational tools exactly reflected the main components of the system: Dialog and presentation, database and inference mechanism. Technically this was an ideal solution. But due to problems with human resources we gave up to implement in Prolog and started to use Delphi for the inference engine too.

The whole system has a clearly object-oriented architecture and was design with strict separation between data keeping, processing and presentation. Due to these design principles adaption to different needs as e. g.

- using national standards and recommendations for time-table presentation,
- adding import filters for track, time-table or rolling stock data,
- implementation of dedicated cost functions or
- changing the inference mechanism

generally will have local effects only.

Remarkable features of the system are:

Application for Rescheduling - The Actual Project together with the Hohenzollerische Landesbahn (HzL)

Promising results of a short test at Dürener Kreisbahn enabled us to establish a pilot project for a greater area. This project is implemented in cooperation with the Hohenzollerische Landesbahn A.G. (HzL) in Hechingen. Compared to the version we used for the first field test in Düren we had to add work in the following fields:

- increasing the reliability,
- improving the user interface (we started with an user interface for the engineer but will finally have to deliver one for the dispatcher),
- improving the modeling of the reality and
- calibration of the program for use in a greater area.

One selected interesting result is shown in figure 2 below. We have two primary delays. 82008 leaves Albstadt-Ebingen at 12.35 (+10min) and 82011 leaves Tübingen at 12.43 (+10 min). Delays remain in the system until 14.36.

Validity of the Model

Our experience from December 2002 to May 2003 showed that our model was practically perfect for the German-Rail-part of the pilot area but we had to add some features to serve some special conditions on the HzL lines. This e.g.

concerned the stations with departure signals only, (goods) trains longer than some station tracks and personnel dependent availability of tracks.

Dimensioning of Rail Infrastructure

During early experiments with our program we got the idea that we can use our software not only to support railway operations but also for planning issues.

We can use the program and one database for the timetable and different databases for infrastructure (before / after removal) and assume some delays. Comparing the results one can easily judge how the absence or presence of infrastructure will influence the stability of operation.

In 1999 DB Netz (German Rail Infrastructure) intended significant reductions of infrastructure be-

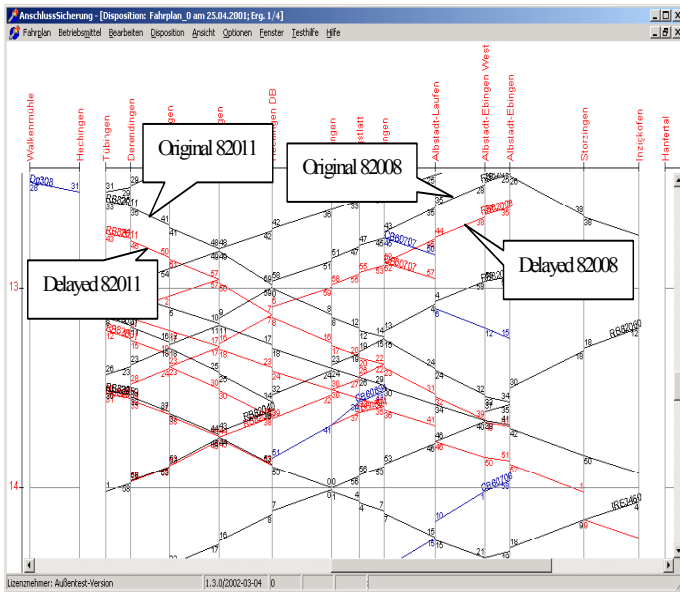


Figure 2: Rescheduling Result for 2 Primary Delays

- It does not implement a node, train, ore line oriented but a totally network oriented optimization of the total situation.
- Train importances (train classes) and importances of connections are respected.
- The importance of a connection may be derived from the classes of trains and the number of passengers involved.
- Unconditional transfer of personnel or rolling stock may be respected.
- Optimization is also possible if there is more than one primary delay in the system.

APPLICATIONS AND EXPERIENCES

Our program actually proved to be useful in two different types of applications:

The original one, which is the rescheduling aid, and a derived one, which deals with the dimensioning of rail infrastructure for the case of delays.

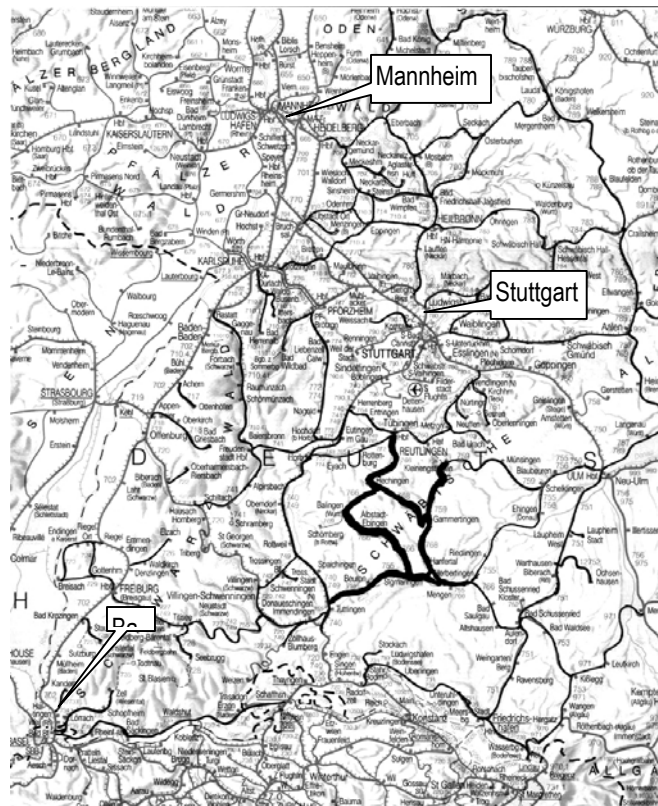


Figure 3: Passenger Railway Network in Baden-Württemberg and Surroundings. (Railway lines are generally shown in grey. Lines where the system could be used directly are shown in black. The pilot area is shown with bold black lines.)

tween Murnau and Mittenwald on the Werdenfels line in Upper Bavaria. Three passing loops and one double track sections which were not used for scheduled traffic were planned to be removed. The plans did not take into account the effect of these removals on delayed trains.

The author was asked for an expert opinion about the effect of these infrastructural reductions on the stability of the timetable. So different infrastructures were investigated in combination with different delays.

The result was that those passing loops etc. which were not used in scheduled traffic brought great advantages in cases of delay and finally - after long negotiations - the original infrastructure was kept. The application of our tool avoided the nearly incalculable deterioration of the traffic situation along that important line.

Introduction of Telematic Concepts

At the moment our rescheduling system is implemented as a stand alone system for manual input and output. But we are sure that adding telematic elements will significantly cut development cost and also greatly increase the operational efficiency. As a result we expect a significant reduction of the total cost of ownership and a large increase in the user benefits.

The telematic elements can be added for different purposes:

- Train and traffic control.
- Passenger information.
- Software maintenance in various aspects.

The telematic system to be built around the rescheduling system as far as possible will use TCP/IP based services.

Presently the rescheduling system exists as a stand alone computer program being operated by one man at one place and the information transfer between the computer and the traffic system is handled manually. However all sorts of higher integration are possible.

In all these systems – stand-alone or integrated – the rescheduling application will not operate safely. Safety critical actions will be either delegated to interlocking systems or to human operators on its own.

Use in other Types of Transport

A modification of our solution for other means of transportation is easily possible. There were some experiments for tramway and bus lines as well as for the German inland waterways.

A test application of our system to control the sequence of ships at the Wasserstraßenkreuz Spandau (waterways crossing Spandau) a bottle-neck in the German inland waterways showed that its use is not restricted to rail respectively land transport.

CONCLUSION

A rescheduling aid as it is developed at the Ravensburg-Weingarten University of Applied Sciences is a versatile and cost effective tool to improve public transportation.

Field trials going on since Dec 2002 showed that we are on a correct and realistic path. RegiDisp could be used on a great number of railway lines in Baden-Württemberg, Germany or abroad (figure 3).

In combination with telematic components there is a great potential for rationalizing operations and improving customer satisfaction as well.

Even if the rescheduling aid RegiDisp is now available as a commercial product there are lots of possibilities and plans for future improvements in the closer area of simulation these are:

- Improved modeling of the trains' movements.
- Computer-aided calibration (data mining) and later even adaptive calibration.
- Improved modeling of infrastructure and operational procedures.
- Getting more experience with other means of transport.

Other work will include

- Optimization (tuning) of the program system in respect to speed and storage needs.
- Further improvements of handling
- Extending RegiDisp for an unlimited area of application.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the funding of this work by the Bundesministerium für Bildung und Forschung (German Federal Ministry of Education and Research) and the Verkehrs- und Umweltministerium Baden-Württemberg (Transport and Environment Ministry of the Federal State of Baden-Württemberg). He also thanks Dipl.-Ing. Bernhard Strobel and Dipl.-Ing. Claus Mohring of the Hohenzollerische Landesbahn AG, Hechingen, Germany for the excellent cooperation.

REFERENCES

- ABB Danmark (Ed) (1994). EBICOS TMS, A New Concept for Integrated Railway Traffic Management. ABB Signal A/S, Hvidovre, Denmark..
- Bär, Matthias (1998). Planungs- und Dispositionssysteme für den Bahnbetrieb - Anforderungen an die weitere Entwicklung rechnerunterstützter Lösungen. *EI-Eisenbahningenieur* (49) 3/98, pp. 19-26.
- Bastin, A. (1996). Telematikunterstützte Informationssysteme für den schienengebundenen Verkehr, *EI-Eisenbahningenieur* (47) 1/96.
- Bertele, M. (2001). Konzept für den Fernservice eines Softwarepakets zur Rechnergestützten Anschluss-

- Sicherung. Sketches for a project thesis, Ravensburg-Weingarten Univ. of App. Science. (not published).
- Fay, A. (1999). Wissensbasierte Entscheidungsunterstützung für die Disposition im Schienenverkehr – Eine Anwendung von Fuzzy-Petri-Netzen, VDI Verlag, Düsseldorf.
- Hanacek, M. (1996). Entwicklung eines Bediensystems für eine rechnergestützte Dispositionshilfe. Diploma thesis, Ravensburg-Weingarten university of Applied Science.
- Hecht, J. (1994). Erstellung eines Dispositionsprogramms für eingleisige Bahnstrecken anhand des Beispiels der Hohenzollerischen Landesbahnen A.G. Diploma thesis, Ravensburg-Weingarten university of Applied Science.
- Koch, W. (1994). Portable elektronische Fahrplanauskunft, Stadtverkehr 4/94, pp. 6 - 8.
- Koch, W. (1995a). Rechnergestützte Dispositionshilfe für eingleisige Bahnstrecken, Proceedings (main volume) 13. Horber Schientage, 22 - 26.11.1995.
- Koch, W. (1995b). Rechnergestützte Dispositionshilfe für den schienengebundenen Regionalverkehr, Proc. (supplementary volume) 13. Horber Schientage, 22 - 26.11.1995.
- Koch, W. (1996). Pünktlicher per Computer, PRO Bahn Zeitung 1/96, Februar 1996, S. 11 - 14.
- Koch, W. (1996). Integrierte Fahrplanerstellung und Konfliktlösung für den Regionalverkehr in der Fläche, Proc. 14. Horber Schientage 1996.
- Koch, W. (1997). Eine kostengünstige Lösung für die rechnergestützte Disposition bei betrieblichen Konflikten - Beispiele und Ausblicke, Proc. 15. Horber Schientage 1997.
- Koch, W. (1998a). Erste Praxiserfahrungen beim Einsatz einer Dispositionshilfe für die rechnergestützte Lösung betrieblicher Konflikte im Regionalverkehr, Proc. 16. Horber Schientage 1998.
- Koch, W. (1998b). Möglichkeiten einer Intelligenten Fahrplankonstruktion, Proc. 16. Horber Schientage 1998.
- Koch, W. (1999). Neue Ansätze für eine Rechnergestützte Netzoptimierung, Proc. 17. Horber Schientage 1999.
- Nuber, G. (1995). Weiterentwicklung eines Programmsystems zur Disposition des Zugverkehrs auf eingleisigen Strecken. Diploma thesis, Ravensburg-Weingarten University of Applied Science.
- Panier, F. (1996). Der elektronische Buchfahrplan. EI-Eisenbahningenieur (47) 1/96.
- Pferdmenges, S.; Schaefer, H. (1995). Automatische Konflikterkennung und wissensbasierte Konfliktlösung in der Streckendisposition. Signal und Draht (87) 10/95.
- Schuster, A. (1992). Erstellung eines Fahrplanauskunftssystems für einen Palmtop-Computer als Zielplattform. Diploma thesis, Ravensburg-Weingarten University of Applied Science.
- Tomii, N. ; Ikeda, H. (1995); A Train Traffic Rescheduling Simulator Combining PERT and Knowledge-Based Approach. Proc ESS 1995, p. 534; URL (withdrawn) www.rtri.or.jp/rd/Train-Traffic/ess_95.html.

BIOGRAPHY

WILFRIED KOCH was born in Heidenheim an der Brenz, Germany. He went to the University of Stuttgart, where he studied electrical engineering and obtained a Dipl.-Ing. (MSc) degree in 1973 and a Dr.-Ing. (PhD) degree in 1980. From 1973 until 1979 he worked as a research and teaching assistant for the university's Institute of Automatic Control and Process automation. From 1979 until 1986 he was responsible for a development project in industrial metrology and head of a development laboratory at Carl Zeiss, Oberkochen. Since 1987 he has been a professor of software engineering and artificial intelligence at the Ravensburg-Weingarten University of Applied Sciences.

Email: koch@fhwgt.de

URLs:

www.stz-sosy.com

erde.fbe.fh-weingarten.de/koch/;

erde.fbe.fh-weingarten.de/koch/english/;

www.stz-sosy.com

Road Traffic And Port Simulation

NUMERICAL SIMULATION OF THE AERODYNAMIC INTERFERENCE IN TRAFFIC FLOW

Marek Maciejewski
Wojciech Osmólski
Instytut Maszyn Roboczych i Pojazdów Samochodowych
Politechnika Poznańska
ul. Piotrowo 3
60-965 Poznań
Poland
E-mail: marekm@sol.put.poznan.pl

KEYWORDS

Road vehicles, vehicle convoy, aerodynamic interference

ABSTRACT

Fundamental purpose of realized simulations was the estimation of the size and importance of aerodynamic interference between vehicles moving in a convoy. In the work, it has been mainly focused on the aerodynamic drag and the analysis of its formation for particular vehicles. Main indicators of interference were the changes of aerodynamic drag coefficients of particular vehicles according to the configuration of a convoy, the distance between particular vehicles and the form of bodies. Quite a large number of simulation results has made the base for realization of extensive comparisons and analyses.

INTRODUCTION

Numerical simulations of the road vehicle aerodynamics are usually focused on consideration of individual vehicles with the purpose of determining the drags and drag coefficients or else defining the concrete effects concomitant with the air flow. Very rarely (Tsuei and Savas 2001, Azim and Gawad 2000), however, there are investigated the aerodynamic interactions between vehicles moving in real conditions, in that also in conditions of increasingly flow rate and traffic density. In such situations, it is necessary to complement the typical studies of individual, isolated vehicles with about more complicated and exacting studies, i.e. the investigations of mutual vehicular influences in the aspect of aerodynamics.

In present work, we have focused on one of interaction categories, namely on that connected with the vehicles moving in the same direction and in one lane. Simultaneously, we have assumed that the vehicles move individually (they are not mechanically coupled) and independently (they are not automatically controlled). One should additionally state that all our considerations refer to the vehicular traffic and aerodynamic interactions in open space, i.e. without artificial constraints in form of real or wind tunnels.

During this realisation, the special attention has been called to the drag forming as consequence of mutual aerodynamic effects. In this context, it has been studied and detailed

analysed the dependence of the aerodynamic drag (with reference to the whole convoy and also to the individual vehicles) upon various convoy configurations described by different distances between particular cars, with regard to body shapes.

Presentation of realised works has been complemented by the description of numerical methods, detailed definition of the research range and simulation conditions, discussion of examined objects and discretizations of computational domains. All obtained results have been supplied through suitable visualisations helpful for expression of conclusions in the range of assumed research tasks.

METHOD OF THE NUMERICAL SIMULATION

The numerical simulation of the aerodynamic phenomena around the vehicles participating in (dense) traffic has been carried out on base of the Navier–Stokes equations for incompressible medium. A disadvantage of such flow description is the lack, in the continuity equation, of a time evolution term. It causes that this equation can be treated only as divergence-free constraint for velocity. The above situation restricts often the solution of the Navier–Stokes equations to different operator (time) splitting methods known, as instance, as the fractional step method, the projection method, the pressure correction method or the pressure–Poisson method. Another possibility is better to couple the incompressible flow equations through the little modification of the continuity equation, and then solving the flow equations with the use of proper methods to so formulated problem. In the case of this strategy, the modification consists in supplementing the continuity equation by an additional term relevant to pressure. According to the form of this term, it is distinguished two fundamental solution techniques known as the penalty method and the artificial compressibility method.

In our numerical simulations, we have used this last approach and the spatial finite volume discretization for unstructured meshes. In this method, the search of solution in computational domain is carried out separately in all cell-centred finite volumes, and the variation of volume-averaged variables results entirely from the equilibrium conditions on cell faces, i.e. from the flux balance defined by the approximate solution of the Riemann problem for reconstructed fields of variables. Solving the above prob-

lem, we have used an upwind scheme referring to the signs of local eigenvalues. The determination of convective and viscous fluxes has given a base for the time integration and computation of variables in particular cells. For integration, we have used the explicit four-stage Runge-Kutta method, and the length of time step has been chosen automatically. In realised simulations, we have abandoned the dual time method because of the very small cell sizes due to the use of adaptive mesh refinement, and thus very small time steps. In consequence of the above, in time runs of aerodynamic drags, it may observe a initial, fast vanishing disturbances without effect on the accuracy of results in further time.

Taking into account the essential influence of mesh arrangement onto accuracy of results, and the fact that the optimal mesh shape can be variable and a priori unknown, the program for numerical flow simulation has been prepared in an adaptive version. It means that the program performs a suitable standard analysis, determines the sub-domains of the mesh where the solution is not sufficiently accurate, and improves the discretization in those regions until the assumed accuracy criterion has been achieved. Thus our adaptive algorithm, in comparison with the standard ones, contains two new elements: an a posteriori error estimator and a mesh refinement procedure. This adaptive procedure changes the mesh density through the bisection of the longest cell faces, and thus also finite volumes, in indicated regions.

VEHICLES, FLOW DOMAINS, DISCRETIZATIONS AND BOUNDARY CONDITIONS

As the objects of our aerodynamic analyses, it has been assumed passenger car models with very simplified bodies and two rear end forms: the hatchback and the squareback. The assumption of so idealised models was of no great importance for numerical simulations, however, it allows to observe some physical phenomena in flow domains more distinctly than in the case of more streamlined shapes.

Though the full aerodynamic simulation of the flow around vehicles requires three-dimensional analyses, nevertheless, in order to carry out the simulations much quickly, less labour-consuming (in the sense of models preparation) and in wider range (in the quantitative sense), we have decided on the two-dimensional simulations. The above is naturally conditioned by the fulfilment of certain geometric requirements regarding the form and symmetry of bodies.

The creation of simplified vehicle models (Fig.1), and then their placement in a properly geometrically defined outer boundary of flow domain has defined the configuration of computational domains. In order to ensure that no correction of drags and also drag coefficients would be needed, a geometric criterion for the outer geometry choice has been applied. In consequence, it has been assumed that the height of computational domains is 12 meters, the length is 60 meters, and the placement of the first vehicle is always identical.

The base for the discretization was univocal defining the edges of computational domains through the nodal points

placed on the profile line of vehicles and the outer outline of flow domain. Exemplary forms of initial discretizations have been presented in Fig.1.

The boundary conditions for each specific computational domain have been defined in accordance with the below specification:

- on the vehicle profile: motionless solid boundary,
- on the left- and right-side of the tunnel: inflow and outflow boundaries, respectively,
- at the top of flow domains: symmetric boundary conditions,
- at the bottom of wind-tunnel: moving (with inflow speed) solid boundary.

For description of various configurations of computational domains, it has been introduced the uniform notation according to the below pattern:

A-B-C-

where the particular symbols (A, B and C) stand:

- A for the profile type (h – the hatchback and s – for squareback),
- B for the distance from the first vehicle to the second one measured front-to-front or rear-to-rear (in meters),
- C the distance between the first and third vehicle, measured as above (in meters).

Finally, we must state that the vehicle speed or the velocity at inlet always was 20 m/s, and the medium properties have been defined as for the air. Control parameters for the solver and adaptive procedure were identical in all computations.

RESULTS FROM AERODYNAMIC SIMULATIONS

In agreement with the research programme, the aerodynamic analyses of vehicles moving in single file have been carried out for various geometric configurations of the vehicle placement in order to identify the effect of the distance between them on the size of aerodynamic drags. Full range of studies has contained several dozen geometric configurations of three vehicles where the least length of convoy (measured between the front of first and last vehicle) has amounted to 20 meters, and the greatest one – to 40 meters. Accordingly, the least distance between succeeding vehicles has amounted to 10 meters, and the largest one – to 20 meters. The length of each vehicle was 5 meters. It has been also introduced special geometric configurations consisting of only one vehicle, for the comparative purposes. In order to avoid dependence of the results and conclusions on the form of bodies, it has been subjected two different body forms (the hatchback and the squareback) to the analyses.

In consequence of mentioned conditions and assumed purposes of the work, during the analyse of results, it has been focused rather on the comparison of aerodynamic drag coefficients (the indirect results of simulations) than the presentation and discussion of direct results. The lasts have included:

- the final forms of discretization meshes showing the occurrence of great and variable gradients of the pressure and velocity components,

- the pressure distribution in computational domains,
- the distribution of the horizontal components of velocity,
- the vertical velocity component distribution.

However, in the present publication (for graphic reasons), it have been placed only some final forms of discretization meshes (Fig.2) for chosen, characteristic computational configurations. Presentation of all direct simulation results (moreover, from different time steps) is also groundless when searching the general rules for the formation of aerodynamic drags. It will have great weight if the analyse of certain local aerodynamic phenomena is carry out.

Unlike the presentation of direct aerodynamic results which concerns always only one concrete case of vehicle configuration and only one of primitive variables (the pressure and velocity components) spread on a discretization mesh, the presentation of overall aerodynamic properties (the indirect results) can refer simultaneously to one or many different vehicle configurations dynamically, in time function. In consequence, the main attention has been given to indirect

simulation results, and particularly to the comparisons of aerodynamic drag coefficients for different vehicles and different vehicle configurations. It prompts us to conclude on the subject of the consequences of aerodynamic interactions between particular vehicles in various vehicular configurations, i.e. their aerodynamic interference.

In general, the aerodynamic interactions between vehicles moving with constant speed in a convoy can be classified in framework of two primary categories, i.e.:

- as the dependencies between particular vehicles of the same vehicle configuration,
- as the dependencies illustrating the changes of aerodynamic properties resulting from the changes of vehicle configurations.

In this last category, the fundamental questions connected with effect of the position of one vehicle, with constant placement two remaining vehicles, on aerodynamic properties of all vehicles treated individually or wholly are contained.

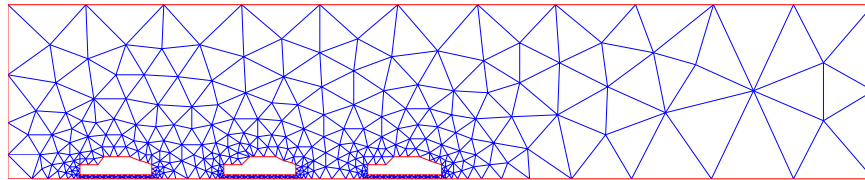


Fig.1. Exemplary initial mesh (the model h-10-20-)

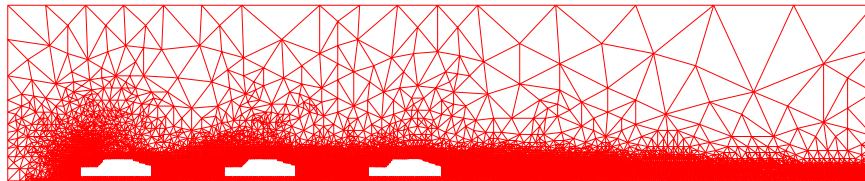


Fig.2. Post-adaptive mesh for the model h-10-20-

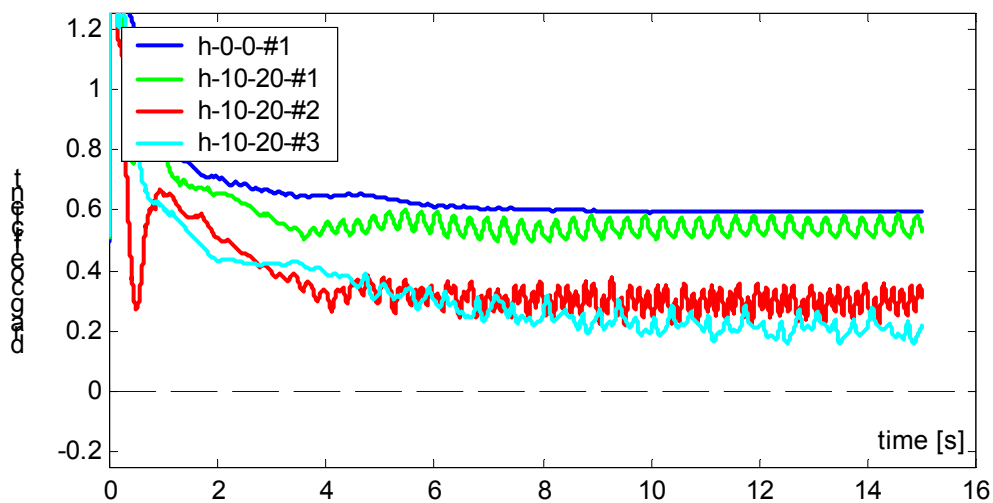


Fig.3. Aerodynamic drag coefficients of vehicles in single file (the model h-10-20-) together with reference to a single vehicle (the model h-0-0-)

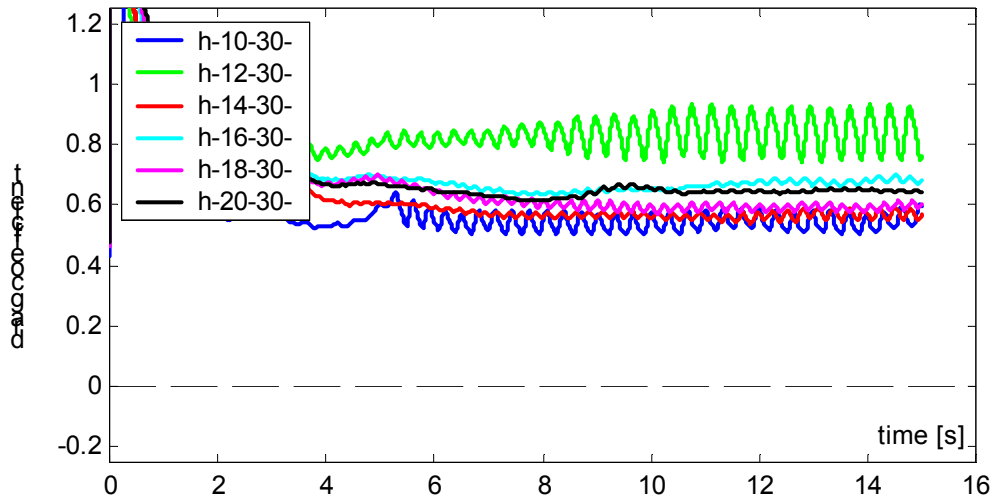


Fig.4. Effect of the middle vehicle position (with constant placement of extreme vehicles) on the drag of the first vehicle

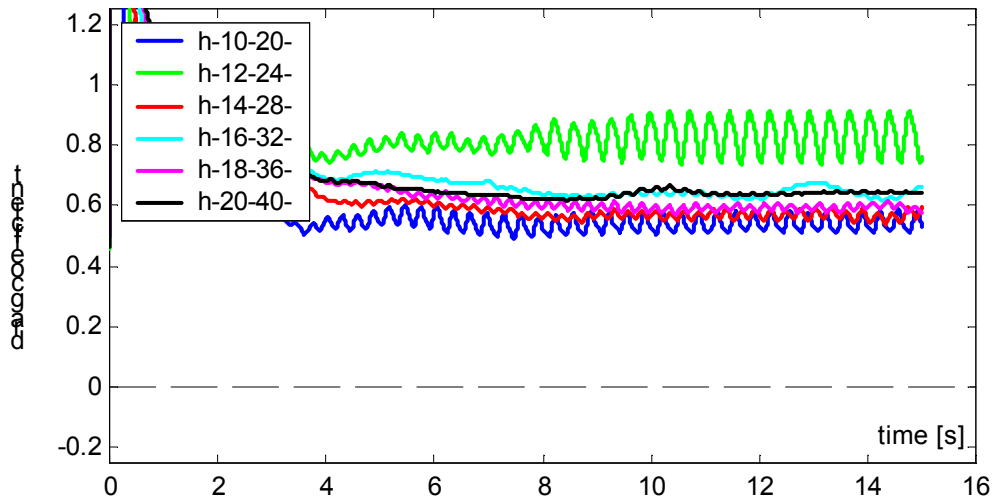


Fig.5. Effect of proportional changes of the distance between successive vehicles on the drag of the first vehicle

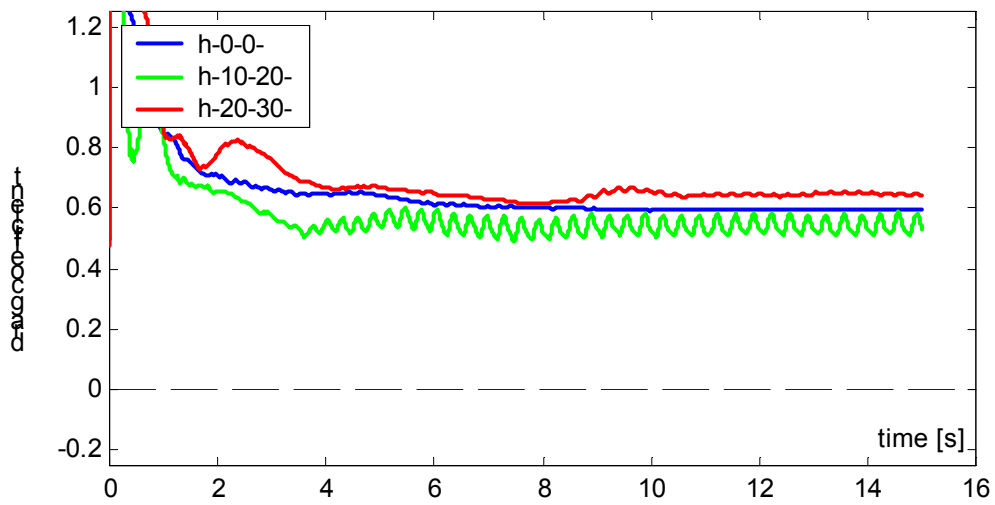


Fig.6. Effect of the position changes of the first vehicle towards the others on the aerodynamic drag of the first vehicle, together with reference to a single vehicle

In present publication, we represent, quite randomly, only some examples of the mutual aerodynamic interference of vehicles. Besides a single example (Fig.3) of the formation of aerodynamic drag for particular vehicles of certain vehicle configuration (together with reference to the drag generated by only one vehicle), all remaining examples (Fig.4 – 6) concern the effect of the different position of second and third vehicle in geometric vehicle configuration on aerodynamic drag of the first vehicle. It is question rather in a less degree sensible intuitively but, as it appears, significant from the point of view of aerodynamics.

Analysing the time runs of aerodynamic drags, we can observe that the first steps (seconds) of simulations are characterised by large drag fluctuations due to the lack of initial conditions in the whole flow domain. Unfortunately, the propagation of boundary conditions in computational space in the case of simplified formulae of time integration takes a time. The whole records of aerodynamic simulation results can be found in the report (Maciejewski et al. 2002).

CONCLUSION

The total and also concise analyse of indirect results, and first of all, the estimation of aerodynamic drag coefficients in time, it was the base for realisation of series observations, and for derivation of more or less general conclusions summarising the aerodynamic aspects of vehicles movement in single file. The most general and important ones have been placed below:

1. Vehicular movement in single file reduces the total amount of aerodynamic drag of all vehicles (the whole convoy), regardless of body forms.
2. The smaller are total distances between vehicles, the smaller is total aerodynamic drag of the whole convoy.
3. Diminishing the distance between two successive vehicles results in the reduction of the aerodynamic drag coefficient for second vehicle whereas the drag of the first vehicle can increase (first of all the hatchbacks) or decrease (mostly the squarebacks).
4. For the single hatchback model, the lack of distinct drag fluctuations is replaced with the greater their occurrence when behind this vehicle are next ones in smaller and smaller distance.
5. Greater, in general, drag fluctuations for squarebacks cause, that even in spite of further growing the fluctuations connected with following vehicles, it can more easily determine general rules governing the average drag levels of successive vehicles for different vehicle configurations.

6. For the hatchback models, the general rules for forecasting the sizes of aerodynamic drag are possible to define, though there are certain specific vehicle configurations which will be not amenable to them.

With reference to the above general conclusions, and especially to that last, it can be noticed few singular vehicle configurations with the hatchback-type bodies which are characterised by specific distribution of aerodynamic drags. Below we mention, as an instance, the two of them:

- for the distance 12 meters between the first and second vehicle, the drag of the first vehicle increases instantly (Fig.4 and 5), and the third ones – drop distinctly,
- for the distances between vehicles 20 and 10 meters respectively (the computational model h–20–30–), the first vehicle drag decreases down to the zero – this rule is not valid for other vehicle configurations where the distance between the second and third vehicle is also equal 10 meters.

With regard to the above, one should remember that those singular aerodynamic behaviours described in the context of mutual vehicle distances, can undergo certain changes together with the change of speed or body form.

REFERENCES

- Tsuei, L. and Ö. Savas. 2001. “Transient aerodynamics of vehicle platoons during in-line oscillations”. *Journal of Wind Engineering and Industrial Aerodynamics* 89, 1085 – 1111.
- Azim, A.F.A. and A.F.A. Gawad. 2000. “A flow visualization study of the aerodynamic interference between passenger cars”. *Vehicle Aerodynamics SP-1524 (AIAA Paper 2000-01-0355)*. SAE, Warrendale, 63 – 73.
- Maciejewski M. et al. 2002. „Symulacja numeryczna aerodynamiki przy ruchu pojazdów w kolumnie”. Research Report 51-845/2002 BW, PP IMRiPS, 1 – 69.

AUTHOR BIOGRAPHY

MAREK MACIEJEWSKI is engaged in two research domains. Both are connected with numerical methods in engineering. The first is geometrically nonlinear theory of thin-walled beams and its finite element realisation, and the second – the solution of viscous, incompressible flows on adaptive unstructured finite volume meshes, with applying to the road vehicle aerodynamics.

AN APPROACH TO MATHEMATICAL MODELLING THE PORT OF SEVILLE

María-José Aracil, José-Ignacio Castillo and Lourdes López-Valpuesta
Department of Economic Theory and Political Economy
Universidad de Sevilla and Centra
41018, Seville
Spain
E-mail: aracil@us.es

KEYWORDS

Harbour Simulation, System Dynamics, Port Economy.

ABSTRACT

The aim of this paper is to explain the functioning of Port of Seville (PS) through a System Dynamics model. PS have experienced an important growth during the last decade, bringing it close to its top capacity. Currently, a significant percentage of Andalusian shore's port traffic, which could eventually choose PS as discharge port, is being deviated to some others competing ports due to PS's lack of capacity. It derives mainly from the small size of vessels PS may admit attributable to insufficient water depth, small width of the lock and the size of the dock.

The simulation model here presented is a first version of a more comprehensive model of PS functioning, intended as a tool for PS Authority to properly choose how to improve PS infrastructure (lock, docks,...) through public and private investments. The model describes, in conventional mathematical language, the impact public investments expected for next years (for enlarging the lock, deepening the openwork and adding length to the dock) may have on coastal traffic.

BACKGROUND –PS LIMITATIONS

The significance of ports on developed economies may be measured through macroeconomic aggregates which quantify this sector's influence on national economies. Specifically in Spain, 59% of exports and 82% of imports are traded through so called ports of *general interest*. In addition to that, 53% of foreign trade with European Union and 96% of trade with the rest of the world is moved through Spanish ports, a distinct proof of the role ports have on Spanish foreign trade.

Our model focusses on PS, the only inland port of *general interest* of Spanish port system, and the single commercial river port in Spain. Located 79 Kms. away from the mouth of the River Guadalquivir, in the economic and social capital city of the Andalusian region, its projection is towards the Atlantic and the Mediterranean.

Though an inland port, its traffic is not fluvial but maritime, so vessels going to or from it come and go to the oceanic

littoral. One of PS main advantages derives from the easy access it provides to production and consumption centres and sites. Additionally, the communications network Seville has developed in recent years provides a fast connexion from the port to its hinterland.

But the inland character of this port also has drawbacks, mainly related to vessels access to it. The port entrance is conditioned by the estuary and by the size of the lock that regulates the water level in the port's commercial area –the water depths of the estuary and the length and width of the lock impose significant size limitations on vessels that may call at PS. Periodical dredges needed to maintain water depths, and lock maintenance costs, increase PS global costs, adding up to tariffs and making it difficult for PS to compete with other ports of Spanish port system which don't have to bear those extra costs.

Together, the facts stated above prevent large-tonnage ships accessing PS, both because of size and costs. Here lies the main cause of a significant traffic deviation to competing ports located at the South-Atlantic littoral.

Because of these characteristics of the port and its access, PS would progressively lose market share and reduce its weight on the total Andalusian traffic (Alatec-Haskoning, 1998).

The aim of our work is to simulate the positive effects public investments may have in PS evolution, helping it to keep its market share and even enlarge it. The building of the new lock and draught will permit larger vessels accessing the port, thus bringing down medium vessels costs (freights, pilotage, stowage, and so on) and increasing PS traffic.

MAIN VARIABLES AND RELATIONS

The simulation model of PS we have elaborated is a medium size model, with 7 level variables, 40 auxiliary variables, and 25 parameters. The model develops interrelations between the following main variables: "Tonnage of Ship willing to berth at PS per year" TONWB, "Length Units of Ship that berth at PS per year" LUB, "Median PS GT", GTMPS, "Length Units of Ship waiting to berth at PS", LUW, "Length Units in Docks", LUD, "Maximum Dock Capacity", MDC and "Goods in Warehouses", GW.

> TONWB: is defined as the difference between each period's rate of traffic and that of former period. The rate of traffic, or entrance traffic, is determined through linear regression between the entrance traffic in the port and Seville (province) income. We justify this relation relying on the foundations of International Trade Theory, in which imports are a function of the importing region income. We have assimilated the concept of imports to that of entrance traffic and supposed the hinterland of the port to equate to Seville province. In our model, we propose this kind of regressions as a useful tool for determining the hinterland of a transport infrastructure.

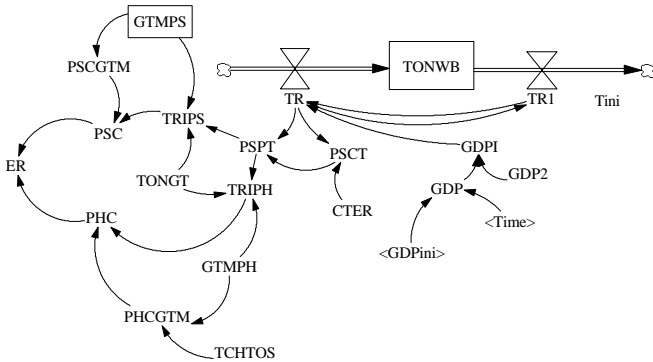


Figure 1. Variable TONWB flow diagram.

> LUB, that is, the Length Units of ship that berth at PS per year, is the most important element considered in the model, as size limitations impede that the true potential traffic PS would be able to berth, actually berth in it. The reason is that it is cheaper to freight just one big vessel that may discharge at, say, Port of Huelva (PH), and then transport goods by road, than freighting two or three smaller ones to berth at PS. LUB is defined as the difference between the rate of Length Units that berth in each period and that of former period. In turn, each period's rate is defined as a function of captive traffic and also of the part of potential traffic which discharges at PS.

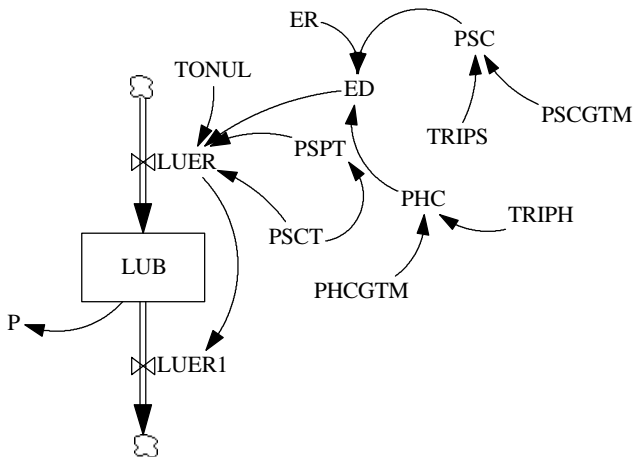


Figure 2. Variable LUB flow diagram.

> GTMPS represents the average GT of vessels calling at PS. As said, one of the main PS problems resides in the impossibility of large vessels entering PS because of their size. One of the units to measure ships capacity which takes into account size is GT. We have focussed on it, for PS, because it is the measure that best represents access difficulties. GTMPS is a level variable that increases with investments and does not decrease.

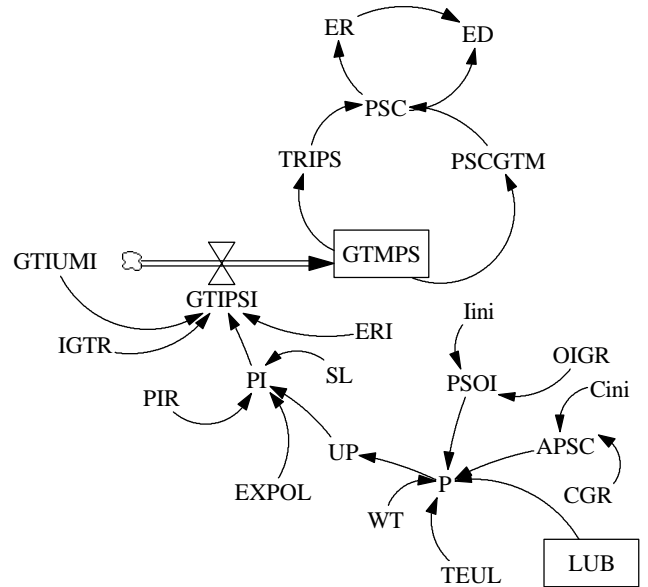


Figure 3. Variable GTMPS flow diagram.

> LUW is the Length Units waiting to berth at PS. Being PS an inland port, and considering the docks limitations, there are time delays in vessels since they get the lock till the berthing at PS. That is why we have defined this level variable, which is related to the dock capacity. It includes vessels that are in their way to be berthed, but haven't reached the dock yet. It is defined as the difference between the Length Units rate that chooses PS for discharging and entrance frequency.

> LUD: this variable represents the part of the dock which is busy at any given time. It is defined as the difference between entrance frequency and exit frequency, and depends mainly on the capacity of the dock, MDC.

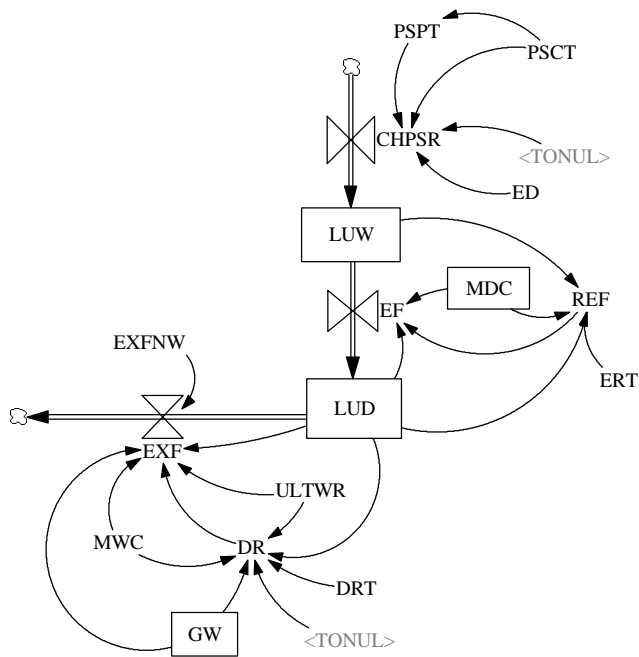


Figure 4. Variables LUW and LUD flow diagram.

γ > MDC is the maximum dock capacity. This variable shows how many length units are available for berthing. It increases with investment, from its initial level. It limits the number of vessels that may berth and, therefore, have to be kept waiting.

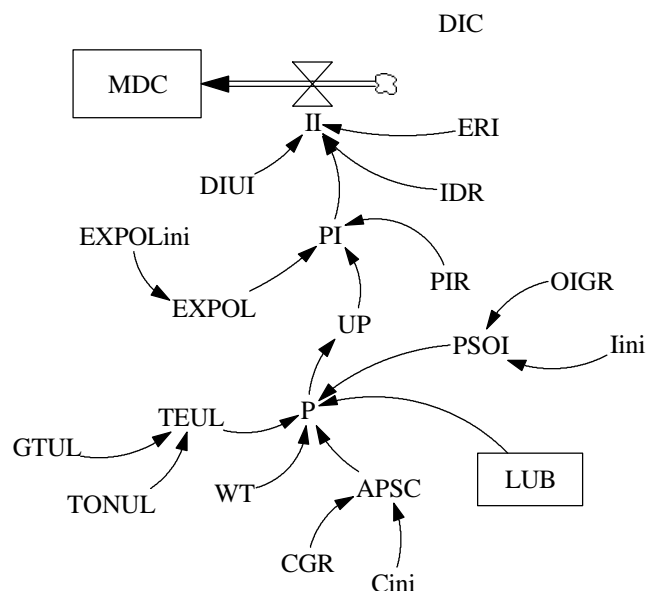


Figure 5. Variable MDC flow diagram.

> GW: all the ships considered entry at PS for discharging merchandise to be stored at PS warehouses, to be stored at third parties warehouses or to be transported by lorries to final destination. This variable would show how warehouses capacities, due to the impossibility of ships discharging if warehouses are busy, might affect traffic through PS. In this case, ships at PS will not leave, thus preventing new entries.

In this version we are not considering this to be the case though.

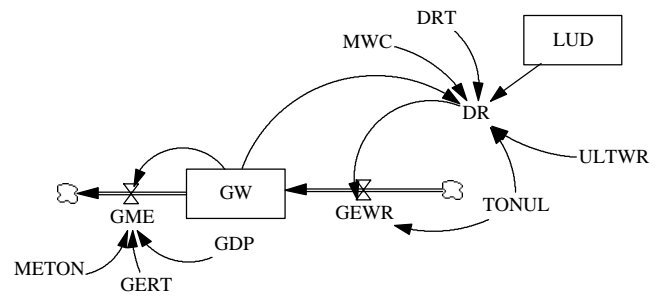


Figure 6. Variable GW flow diagram.

CHOOSING PS

The main problem PS experiences is the loss of potential traffic, hindered by its infrastructure limitations.

We have calculated total entrance traffic, and compared costs this traffic would have to pay both in PS and in a competing port. As competing port we have chosen PH as it receives most of the traffic that does not discharge at PS, even though goods are to be transported to Sevilla. The reason for choosing PH is related to its infrastructure, that allow larger vessels to entry, and to its proximity and easy merchandise transportation to Seville by highway.

Choosing PS depends mainly on each's period costs comparison of PS and PH (Port of Seville Authority, 1995):

- Highway transport costs from Huelva to Seville is included in PH costs.
- PH has been chosen as the competing port because in fact PH average GT is much higher than PS's. This leads to smaller average costs for PH.
- Public investments increase Seville's average GT.
- PH's average GT is considered to be a constant.

If entrance traffic costs are altogether cheaper for PS than for PH, PS will be chosen and PH will be discarded. If entrance traffic costs were smaller at PH, choosing PS will follow a ratio defined as an inverse proportion of the relative difference between PS and PH costs –if PS costs are smaller than 110% of PH costs, 90% of the traffic would choose PS; if PS costs are smaller than the 120%, 80%, and so on.

When determining the traffic rate that discharges at PS, a distinction is drawn between captive traffic and potential traffic. The reason is that around 50% of the PS traffic is not related to costs, that is, those vessels would discharge at PS in any case (Port of Seville Authority, 1999a). The rest of traffic, so called potential traffic, is the only one affected by the choosing PS decision described.

The decision will not be static, since Seville's GT will be increased by public investments, being higher the percentage of the traffic that will eventually choose PS, since costs will decrease, relative to PH, as a result from those investments.

In this first version we are mainly considering investments to affect GT and because of this, the size of vessels that may discharge at PS, though investment may also affect dock size. Since PS is almost at its top capacity and bottle necks may occur at the berthing point, docks capacity should also increase by investments in order to allow larger ships to berth.

The sources for financing investments would basically be:

1. FEDER (external financing).
2. Selling obsolete PS land.
3. Internal financing (profit).

Internal financing is in the origin of the model feed-back because the larger volume of traffic the new infrastructure will allow will increase the tariffs income. Which, in turn, will increase profit and the part of it set aside for new investments.

INITIAL ASSUMPTIONS

The model is based on the following assumptions:

1. We have considered just entrance traffic, that is, ships discharging goods, not the charging of them.
2. Province of Seville has been considered to be the hinterland.
3. Since port traffic in ships is not a continuous variable, we had to make some adjustment. First, while having to choose PS or PH as the cheapest port, making the costs comparison for a medium ship, it was necessary to adopt a criterion for choosing this ship –Ton., GT, length, etc. When evaluating dock limitations, we chose length units.
4. Taking as a reference last 10 years data, we have applied a 0,07% rate of growth to costs and 0,08% to income (Port of Seville Authority, 2003).
5. GDP initial growth is fixed at 3%, though alternative scenarios will be considered.
6. In this version, strangling in the warehouses is not considered.
7. No difference is made regarding kind of merchandise (liquid bulks, solid bulks, general merchandise, containers), nor type of vessel (bulk carrier, container ship, etc).

8. No specializing of wharfs neither especial infrastructures for them are considered when distributing goods traffic. On the contrary, a unique wharf is considered.

9. PH has been taken as the solely competing port.

REFERENCES

- Alatec-Haskoning. Ingenieros y Arquitectos Consultores. 1998. *Asistencia Técnica para el Estudio de alternativas de las obras para la modificación del acceso al puerto e incidencias previstas de tales obras en el tráfico, la operación portuaria y zonas de servicio.*
- López Díaz-Delgado, E., Martínez Vicente, S. 2000. *Iniciación a la simulación dinámica. Aplicaciones a sistemas económicos y empresariales.* Ariel Economía.
- López Valpuesta, L., Castillo Manzano, J.I. 2001. *Análisis de la actividad económica del puerto de Sevilla y su influencia provincial.* Servicio de Publicaciones de la Universidad de Sevilla, Autoridad Portuaria de Sevilla. Seville.
- Port of Seville Authority, "Estudio de la Estructura de Costes para la captación de tráfico en el Puerto de Sevilla". September, 1995.
- Port of Seville Authority, *Anual Port of Seville Authority Report*, 1990 to 2002.
- Port of Seville Authority. 2003. *Business Plan.*
- Port of Seville Authority. 1999a, *Development Plan.*
- Port of Seville Authority. 1999b. *Strategic Plan.*
- Salini, P. 1999. *La dynamique des systèmes: Retour sur une expérience d'application aux transports de marchandises.* <http://patrice.salini.free.fr/Dynami.pdf>
- Sterman, J. 2000. *Business Dynamics*, Irwin-McGraw-Hill.

BIOGRAPHIES

MARIA-JOSE ARACIL was born in Madrid. She obtained a degree in Economics in 1994, and a Ph.D. in 2001, at the Universidad de Sevilla. She worked at the Commerce and Industry Regional Government. Nowadays she lectures at the Economic Theory Department, where she is doing research on System Dynamics applied to social sciences.

JOSE-IGNACIO CASTILLO was born in Sevilla. He obtained his degree in Economics in 1995, his degree in Business Administration in 1997 and his Ph. D. in 1999, at the Universidad de Sevilla. After working at Escuela Universitaria de Osuna and Unicaja, he has been lecturing at the Economic Theory Department where he is doing his research on Port Economy. He is Vicedean for Quality and Academic Affairs.

LOURDES LOPEZ-VALPUESTA was born in Sevilla. She obtained her degree in Economics in 1995 and her Ph.D. in 2000, in the Universidad de Sevilla. After completing a research fellowship, she has been lecturing at the Economic Theory Department where she is doing her research on PS impact on Seville's economy.

**MODELLING
AND
SIMULATION
IN
THE
TEXTILE
INDUSTRY**

Textile Processes

SIMULATION AND CONTROL OF PACKAGE DYEING PROCESS BASED ON A MATHEMATICAL DESCRIPTION OF DYE TRANSFER THROUGH THE PACKAGE

Renzo Shamey
Xiaoming Zhao
Marcus Vosoughi
School of Textiles & Design
Heriot-Watt University
Scotland, UK, TD1 3HF
r.shamey@hw.ac.uk

KEYWORDS

Mathematical Modelling, Package Dyeing, Adsorption, Simulation, parameter identification, differential equations, control systems, real time

ABSTRACT

A mathematical model of the package dyeing process, based on the physico-chemical laws of dyeing that govern transfer, adsorption, and diffusion of dyes, is established in this study. The model is used to simulate various control parameters (such as temperature, flow rate and direction, etc.) and to isolate the variables that affect the levelness of dye across the package. The effect of flow direction on the overall rate of dye exhaustion and levelness is described. The relationship between various profiles used to inject dye into the dyebath and the degree of levelness achieved is also demonstrated. The control strategy adapts the right-first-time strategy, which minimises the environmental impacts of the package dyeing while reducing the consumption of dyes, chemicals, water and energy with immediate benefits to both the dyer and the consumer.

1. INTRODUCTION

Package dyeing of many types of yarn in batch re-circulatory machines has attracted considerable attention in recent decades and is expected to grow further in future due to the special characteristics of package dyeing process (Nobbs 1985). However, there are problems within the process, including the uneven distribution of dye along the radius of the package, which impact on the overall cost of dyeing.

In order to elucidate these factors systematically, an investigation of various parameters that influence the outcome of the dyeing process, such as the type of dyes and

auxiliaries used, type of machinery and the effect of dyeing conditions on levelness, is required (Shamey and Nobbs 2000a; Shamey and Nobbs 2000b).

It was recognised that because of the large number of the variables involved, it would be difficult to determine by experiment the influence of an individual variable on the results of a dyeing (Boulton and Crank 1952; Hoffman 1979).

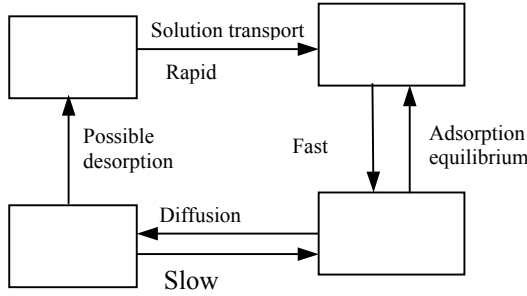
Laboratory scale investigations provide some insight into the effect of certain variables, but cannot reveal the interaction of the various variables clearly. There are commercial dyebath control systems available, but these are usually only designed to follow a simple time-temperature profile (Gilchrist 1995).

Several rigid models (Hoffman 1979; Wai 1984; Vosoughi 1993) have been applied in package dyeing to successfully describe the mechanisms of adsorption and diffusion phenomena, however, the mathematical complexity of these models makes them rather inconvenient for practical use.

A mathematical description of the dyeing process based on the physico-chemical laws is therefore necessary in order to get some insight into the basic processes operating inside the package dyeing machine and to isolate the variables.

2. THE KINETICS OF DYEING

Dyeing is a process, which takes time. The transfer of a dye molecule from the dye solution into a fibre is usually considered to involve the initial mass-transfer from the bulk solution to the fibre surface, adsorption of the dye on the surface, followed by diffusion of the dye into the fibre (Scheme 1) (Broadbent 2001).



Scheme 1

The adsorption equilibrium is rapid, and it is usually assumed that the overall rate of dyeing depends on the rate of diffusion of the dye into the fibre. Inadequate control of the rate of dye adsorption will obviously result in unlevel dyeings unless the dye can subsequently migrate from deeply dyed to lightly dyed regions of the package (Vickerstaff 1950). Therefore, the control of the first two stages of the process, namely the initial mass-transfer from the bulk solution to the fibre surface, and adsorption of the dye on the surface, is important for a level dye distribution throughout the package to be achieved.

3. THE THEORETICAL MODEL

3.1 General Assumptions

In order to keep the model as simple as possible, the following assumptions are made:

1. The dye-liquor is assumed to circulate through the package, the direction of the flow being perpendicular to the layer. The package is represented by an infinitely extended plane parallel layer of thickness l (Figure 1).
2. The dye liquor is well stirred so that the concentration of dye across the entrance of the package throughout the liquor can be taken to be uniform as C_1 .
3. Package density and flow velocity inside the package are assumed to be uniform, i.e. deformation of the package, and the influences of the shape of the package on liquor flow, are not considered.
4. The flow is assumed to be laminar, characterized by smooth streamlines.
5. The dye concentration at the surface of the fibre is related to the dye concentration in the liquor within the package (local dyebath concentration) according to the sorption isotherm of the dyeing equilibrium.
6. The dye adsorption and any chemical reactions with fibres occur very rapidly in comparison to the diffusion process, thus local equilibrium can be assumed to exist between the free and immobilized components of the diffusing substance.
7. The temperature dependence of the diffusion coefficient is represented by the Arrhenius Equation (Johnson 1989).
8. Deviations from linear liquor flow, which are caused by the fibres' hydrodynamic boundary layer, are neglected.

Thus, there is no need to consider the arrangement and orientation of the fibres within the package.

9. The temperature within the entire system including the package is uniform.

These assumptions are reasonable as the package dyeing machinery are well stirred and the control of the temperature within the system is now routine. Although it is assumed that the flow direction is perpendicular, the dispersion of the liquor within the package is taken into account.

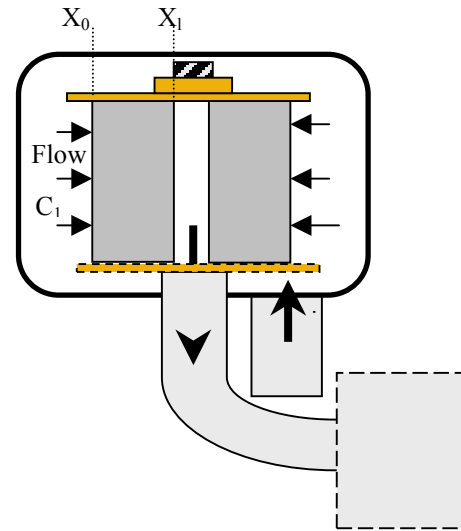


Figure 1: Schematic Representation of the Package

3.2 Mathematical Model

The dispersion of chemicals in liquor is often presented as the sum of the contributions of molecular diffusion and flow (convection). In dyeing, when the dye solution crosses the yarn package, a chemical transport in a porous media occurs. Meanwhile, during this process the dyes may be adsorbed by fibres, and this phenomenon is assumed to decrease the rate of dye transfer through the geometry of the system. If the package density and flow velocity inside the package are assumed to be uniform and independent of time, the combined Convection/ Diffusion equation with the inclusion of an Adsorption factor can be used to simulate the convective dispersion of dyes' transfer process in packed-bed reactors (Wai 1984) as shown in Equation (1).

$$\frac{\partial C_p}{\partial t} = D_f \frac{\partial^2 C_p}{\partial x^2} - V \frac{\partial C_p}{\partial x} - \lambda \frac{\partial C_f}{\partial t} \quad (1)$$

Where D_f , V and λ represent the diffusion coefficient, flow velocity and the adsorption factor respectively.

The initial condition for Equation (1) may vary according to the system under consideration. When the yarn is initially

undyed and the dye is introduced at time $t = 0$, the following condition will apply:

$$C_p(x, 0) = 0, \text{ for } 0 < x < l. \quad (2)$$

In order to solve these equations, proper boundary conditions have to be introduced. If the concentration of dye in the liquor at the entering stream is considered to be C_1 , the concentration just within the entrance of the packed bed will be less than C_1 due to the dispersion in the axial direction (Wai 1984). Therefore, the rate at which dye crosses the entrance plane $x = 0$ as a result of combined bulk flow and dispersion, may be represented by Equation (3):

$$D \frac{\partial C_p(x_0, t)}{\partial x} + VC_p(x_0, t) = VC_1 \quad (3)$$

On any other surface of the package, the transfer of dye along the length of the package by dispersion is negligibly slow compared with the rate of mechanical flow.

The exit boundary condition can be defined as:

$$\frac{\partial C_p(x_l, t)}{\partial x} = 0 \quad (4)$$

Equations (1), (2), (3) and (4) generally describe the dye dispersion process with flow through the package in one dimension.

Where:

- C_1 The concentration of dye in the mixing tank (g/dm^3),
- V The flow velocity of dye liquor before the entry line of the package,
- $C_p(x, t)$ The concentration of dye in the liquor within the package at distance x from the entry line and time t (g/dm^3),
- $C_f(x, t)$ The concentration of dye on the fibre at distance x from the entry line and time t (g/dm^3),
- t Time (s),
- $M(t)$ Amount of dye within the package at time t (g),
- P_d Package density (g/cm^3),
- T Temperature (k),
- D_f Molecular diffusion coefficient,
- l Package thickness.

If we consider that adsorption accompanies the dispersion process, the relationship between the C_f and C_p can be defined as shown below:

$$C_f = K_F C_p^n \quad (5)$$

The substitution of C_p from Equations (1), and (5), and the use of assumptions 4, 5, and 6 (Cussler 1997, Crank 1975), will result in Equation (6):

$$\frac{\partial C_f}{\partial t} = \frac{\partial}{\partial x} \left(\frac{D_t}{n} \left(\frac{1}{K_F} \right)^{\frac{1}{n}} C_f^{(1-n)/n} \frac{\partial C_f}{\partial x} \right) \quad (6)$$

When $n = 1$, Equation (6) can be simplified as:

$$\frac{\partial C_f(x, t)}{\partial t} = \frac{\partial}{\partial x} \left(\frac{D_t}{K_N} \frac{\partial C_f(x, t)}{\partial x} \right) \quad (7)$$

When $n = 1/2$, Equation (6) can be simplified as:

$$\frac{\partial C_f}{\partial t} = \frac{\partial}{\partial x} \left(\frac{2D_t}{K_F^2} C_f \frac{\partial C_f}{\partial x} \right) \quad (8)$$

If the relationship between C_f and C_p obeys the following form (Langmuir isotherm):

$$C_f = \frac{K_L C_p}{1 + K_L C_p} \quad (9)$$

the substitution for C_p from Equations (1), (9), and assumption 4, 5, and 6, will result in:

$$\frac{\partial C_f}{\partial t} = \frac{\partial}{\partial x} \left(\frac{D_t}{K_L (1 - C_f)^2} \frac{\partial C_f}{\partial x} \right) \quad (10)$$

Where D_t is the temperature and flow dependent dispersion coefficient across the thickness of the package, and K_N , K_F , K_L represent sorption coefficients for Nernst, Freundlich, and Langmuir isotherms, respectively.

Equations (7), (8) and (10), with proper initial and boundary conditions 2, 3, and 4, can be used to simulate the dye dispersion process through the package in the case of Nernst, Freundlich, and Langmuir adsorption isotherms, respectively.

It should be noted that unlike D_f in Equation (1), D_t is a temperature and flow dependent dispersion coefficient, which can be described as

$$D_t = f(T, V)$$

It has a very different physical origin and proceeds at a different speed compared to the diffusion process.

4. NUMERICAL RESULTS

The common parameter values used to solve the general equation are given below:

Package (Ren 1985):

Inner radius	0.18 dm
Outer radius	0.56 dm
Height	1.36 dm
Thickness (l)	0.38 dm

Equation (7) was solved using Matlab software package under selected conditions. The results are mainly presented in graphical forms.

Figure 2 shows that after a short dyeing time, dye distribution factor $C(x_i, t)/C(x_0, t)$ increases rapidly at the beginning of the dyeing time, indicating a rapid dye transfer through the package during this period; and then, becomes smaller with time especially after half levelling time.

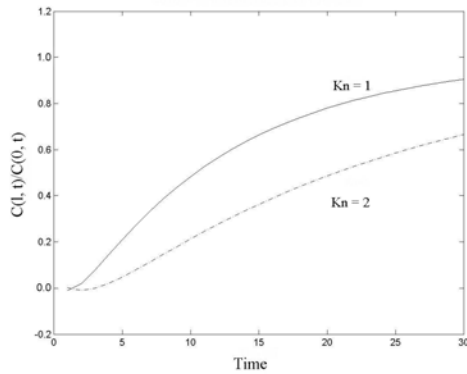


Figure 2: Comparison of Dye Distribution When $Kn=1$ and Kn is Greater Than One

Dye dispersion through the package will take a considerably longer period of time if the sorption and reaction between the dye and fibre during the process are taken into account (in this case, $Kn = 2$) compared with the condition of ignoring the sorption factor ($Kn = 1$).

Temperature, pH, flow rate, and some other factors affect the dyeing process by influencing the overall dispersion coefficient D . The distribution factor is defined by $L(t)$ which represents the numerical value of $C(x_i, t)/C(x_0, t)$. We have used an arbitrary time when $L(t)$ reaches 95% (t_{L95}) to represent a sufficient distribution of dye across the package. A comparison of $L(t)$ against the change of D , can produce the theoretical effect of different values of D on the levelling time (see Figure 3).

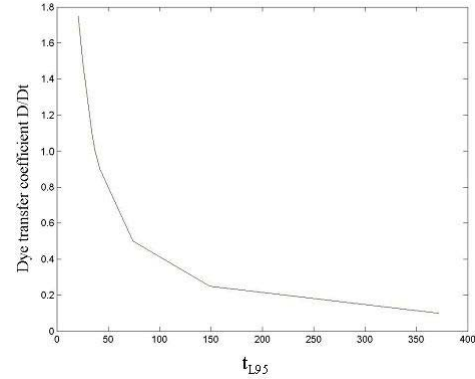


Figure 3: Variation of D Against t_{L95}

It can be seen from figure 3 that t_{L95} decreases with an increase in D . When the change in D/D_i around unity is approximately 50%, the levelling time does not change significantly. However, if D/D_i decreases further, only a small decrease can cause a much longer levelling time. The effect of D on t_L is more complicated, since levelness depends on D in two ways: a higher D leads to a more rapid dye migration within the package, but also to a higher initial unlevelness.

Figures 4, and 5 are visualised solutions of Equation (7), and illustrate the distribution of dye concentration on fibres within the package in both cases of flow direction. It should be noted that under the same conditions $C(x_i, t)/C(x_0, t)$ for the out-to-in flow direction is 62%, while for the in-to-out direction, only 36% is achieved.

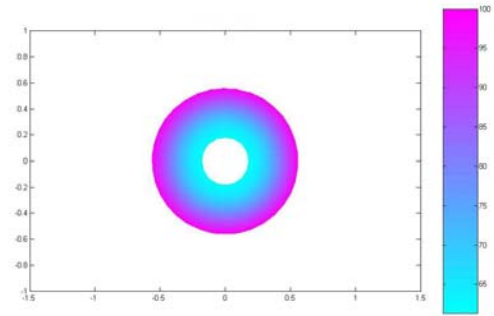


Figure 4: Dye Distribution for the Out-To-In Flow Direction

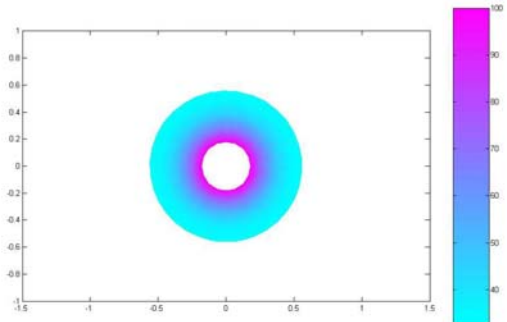


Figure 5: Dye Distribution for the In-To-Out Flow Direction

During the dyeing, the concentration of dye in the mixing tank (C_1) can be controlled by dosing the appropriate quantity of dye to the system. If C_1 is a function of time, it causes a variable boundary condition at line X_0 . Under this condition, the solution of the model can be used to evaluate different dye dosing profiles. This can be used to demonstrate a suitable control strategy for the dyeing process. The increase of C_1 with time can be controlled according to various profiles, including: Linear, Exponential, and Quadratic. The dye distribution factor within the package against the dyeing time for these profiles can be seen in figure 6:

It can be seen from figure 6 that the trends of the dye distribution factor $L(t)$ represented by $(C(x_i,t)/C(x_0,t))$ against the dyeing time are similar between the three dosing methods. However, the quadratic profile (dashed line) demonstrates a sharper increase in $L(t)$ during the first 15 time units of dosing, while the linear profile (continuous line) and the exponential profile (dotted line) produces a gradual increase in $L(t)$ throughout the dosing time. This infers that, if the quadratic dosing algorithms are employed, only within a short period of time after the start of dosing, a fast dye migration throughout the package can be achieved, and then, the dosage makes a small impact in dye distribution, and therefore the control of levelness may depend on other factors such as temperature, pH, etc.

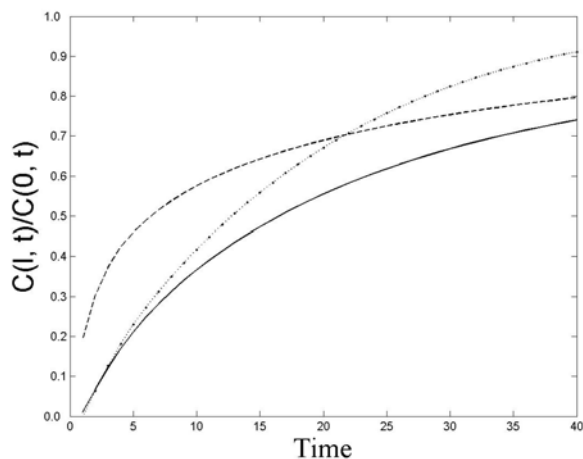


Figure 6: Relationship Between Dye Distribution Factor and Various Dosing Profiles

The comparison of the distribution factor after 40 time units indicates that using the exponential dye injection profile (dotted line) produces a better dye distribution ($L_t=91\%$) across the package, compared to the linear and quadratic dosing methods which only achieve 79%, and 74% levelness respectively. This indicates that if there is a linear relationship between C_p and C_r , (Nernst isotherm) an exponential dosing profile will be the preferred method to

obtain a level dyeing. This, of course, needs to be verified by further experimental works.

5. CONCLUSIONS

The mathematical model presented here offers the possibility of describing the dye transfer process, based on dispersion, convection, and various adsorption isotherms. The influence of process parameters, such as dyeing temperature, flow rate, flow direction, dye/fibre type, and dye concentration in the mixing tank on the outcome of the dyeing process can be examined. The relevant adsorption isotherms can be selected for different dye/fibre combinations, to simulate the dye dispersion process through the package. A dye distribution factor L_t , which can represent the levelness, is defined and the relationship between selected control parameters and L_t are demonstrated. A comparison of dye dosing profiles and their effect on the distribution of dye across the package indicates that for a Nernst dye-fibre isotherm an exponential dosing profile would produce more level results.

REFERENCES:

- Boulton J, Crank J. 1952, "Package Dyeing: Theoretical Model and its Relation to Technical Practice." *Journal of the Society of Dyers and Colourists*, Vol. 68, p 109.
- Broadbent A.D. 2001, *Basic Principles of Textile Coloration*, Society of Dyers and Colourists, Bradford, UK.
- Crank J. 1975, *The Mathematics of Diffusion*, Oxford, UK
- Cussler E. 1997, *Diffusion: Mass Transfer in Fluid Systems*, 2nd edition, Cambridge, UK.
- Gilchrist A., 1995, "Control of Dyeing Processes by Colour Measurement." *Review of Progress in Coloration*, Society of Dyers and Colourists, Vol. 25, p 35.
- Hoffmann F. 1979, "Model Calculations on Levelness in Package Dyeing." *Journal of the Society of Dyers and Colourists*, p 178.
- Johnson A., 1989, *The Theory of Coloration of Textiles*, SDC, Bradford.
- Nobbs J.H, Ren J., 1985, "Computer Control of Batchwise Dyeing Process." 29th Hungarian Textile conference, Budapest.
- Ren J. 1985, PhD. Thesis, Leeds University, Leeds, UK.
- Shamey R.; Nobbs J.H., Feb 2000, "The Application of Feed-Forward Exhaustion Profiles in The Control of Dyeing Machinery." *Textile Chemist and Colorist & American Dyestuff Reporter*, Vol. 32, No 2, pp 47-52.
- Shamey R., Nobbs J.H., June 2000, "The Use of Colorimetry in the Control of Dyeing Machinery." *Textile Chemist and Colorist & American Dyestuff Reporter* Vol. 32, No 6, p 32-37.
- Vosoughi M. 1993, PhD Thesis, Heriot-Watt University.
- Vickerstaff T. 1950, *The Physical Chemistry of Dyeing*, Oliver and Boyd, London.
- Wai P. 1984, PhD Thesis, Heriot-Watt University.

AUTHOR BIOGRAPHY

RENZO SHAMEY is the head of Automation and Novel Coloration (ANC) Research Group at the School of Textiles of Heriot-Watt University. He obtained his PhD from The Department of Colour Chemistry of Leeds University in 1997. He joined Heriot-Watt University in 1998 as Lecturer in Dyeing, Printing and Finishing.

XIAOMING ZHAO is an associate professor at Xinjian University in China. He is currently a member of ANC research group at Heriot-Watt University.

MARCUS VOSOUGHI is a lecturer in IT and computer science at the School of Mathematics and Computer Sciences of Heriot-Watt University.

MODELLING DYE LIQUOR FLOW IN FABRIC BEAM DYEING AND YARN PACKAGE DYEING

David Karst and Yiqi Yang
University of Nebraska
234 HE Bldg
Lincoln, NE 68583-0802 USA
E-mail: dkarst1@bigred.unl.edu

W. Alex Rapp
Institute of Textile Technology
2551 Ivy Rd
Charlottesville, VA 22903 USA

KEYWORDS

Computational Fluid Dynamics, fabric beam dyeing, yarn package dyeing, dye levelness, porous media

ABSTRACT

Level dyeing is difficult to obtain in fabric beam dyeing and yarn package dyeing because the fiber remains stationary while the dye liquor is pumped through it. The liquor flow profile is critical to obtain level dyeing because each section of the package must receive sufficient liquor flow. This research used Computational Fluid Dynamics (CFD) to model dye liquor flow during beam dyeing and package dyeing and to determine how certain parameters affect liquor flow through the packages. The liquor flow profiles from CFD agreed with K/S measurements made on dyed fabric. This research found that winding beams tightly improved liquor flow uniformity, and winding beams or yarn packages tightly in the inner layers and more loosely in the outer layers further improved flow uniformity. In beam dyeing, liquor flow uniformity can be improved by placing collars under the fabric a distance of about 30% of the winding height on both sides of the fabric or by placing end-caps against both sides of the fabric beam. For low density fabrics, the liquor tends to shortcut through the fabric and flow directly to the drain, but the beam can be wrapped with a low permeability material to improve flow uniformity.

INTRODUCTION

In fabric beam dyeing and yarn package dyeing, it is difficult to evenly apply the dye to the fiber because the textiles remain stationary during dyeing. The flow profile of dye liquor through the fabric beam or yarn packages is an important factor in determining the levelness of dyeing. However, the liquor flow profile cannot be measured directly, but a model is one way to study the liquor flow indirectly.

One tool available to model liquor flow in fabric beam dyeing and yarn package dyeing is Computational Fluid Dynamics (CFD), which allows for analysis of fluid flow direction and magnitude under various conditions (Farber et al. 1999). This research utilized CFD to model how certain variables affect dye liquor flow during fabric beam dyeing and yarn package dyeing. The goal was to determine how to promote uniform flow of the dye liquor during fabric beam dyeing and yarn package dyeing so that level dyeing can be obtained. Previous research showed that increasing dye liquor flow through a fabric

beam as calculated by CFD corresponded to increasing K/S values measured on fabric dyed under the same conditions, which indicated that CFD is a valid tool to model dye liquor flow in beam dyeing and package dyeing (Rapp et al. 2002).

EXPERIMENTAL

CFD was used to model how certain variables affect dye liquor flow during fabric beam dyeing and yarn package dyeing. The effects that were studied include the fabric beam and yarn package density profile, collar placement, end-caps, and outer beam wrap. To create the flow models, data were obtained from actual beam dyeing and package dyeing processes. Besides the physical dimensions of the equipment, beam, and packages, the permeability constants of the fabric beam and yarn packages were obtained to characterize the interaction between the dye liquor and fiber. The amount of friction between the dye liquor and fiber increases with decreasing pore size, which decreases the permeability of the fabric or yarn (Kulichenko and Langenhove 1992). To calculate the permeability constants α in m^2 , D'Arcy's Law was applied in Equation (1), which was valid since the liquor flow was laminar through the textiles (Gibson 1999). A Kretschmer Textiluhr permeability tester was used to measure the time t in s required to pull $1.135 \times 10^{-3} \text{ m}^3$ of air, V , with a viscosity μ in $\text{Pa}\cdot\text{s}$ through a sample with a thickness Δx in m and a cross-sectional area A in m^2 at a pressure drop ΔP of $\sim 200 \text{ Pa}$.

$$\Delta P = -\frac{\mu}{\alpha A} \frac{V}{t} (\Delta x) \quad (1)$$

The data collected from the actual dyeing processes were used to create the liquor flow models by using CFD software available from Fluent, Inc. (Fluent 2000). For the fabric beam dyeing model and the yarn package dyeing model, the kier, the fabric beam, and yarn packages were isolated as the flow regions of interest. The yarn package dyeing model was based on a single spindle sample package dyeing machine. Using the CFD software, the flow region was divided into control volumes called a mesh, which had a resolution of 8.0 cm^3 and 1.0 cm^3 for the beam dyeing model and a resolution of 0.26 cm^3 for the package dyeing model. At each control volume, the CFD software used Equation (2) to calculate the magnitude and direction of the liquor velocity. In Equation (2), t is time, V is the volume of the control volume, ρ is the density of the fluid medium, A is the cross-sectional area perpendicular to the direction of flow, v is the velocity of the fluid medium, Γ is the rate of diffusion of

mass of the fluid medium per unit cross-sectional area, S is the rate of generation of mass of the fluid medium per unit volume, ϕ is 1 for the equation of continuity or mass, ϕ is u for velocity for the equation of x -momentum, and ϕ is v for velocity for the equation of y -momentum.

$$\frac{\partial}{\partial t} \int_V \rho \phi dV + \oint_A \rho \phi v \cdot dA = \oint_A \Gamma \nabla \phi \cdot dA + \int_V S \phi dV \quad (2)$$

From the liquor flow profiles calculated with CFD, the percent flow differences were determined and indicated the uniformity of the liquor flow profiles. For the beam dyeing model, the side-center percent flow difference was the percentage difference between the average liquor velocities at the center and sides of the fabric in the outer fabric layers of the beam. The top-bottom percent flow difference was the percentage difference between the average liquor velocities in the outer (top) and inner (bottom) fabric layers of the beam at the center of the fabric. For the package dyeing model, the percent axial flow was determined, which was the axial component of the liquor velocity as a percentage of the magnitude of the liquor velocity. The axial direction is parallel to the axis of a yarn package, and radial direction is parallel to its radius.

The effect of overall fabric beam density on the dye liquor flow profile was determined for densities of 550, 595, and 637 g/L at a constant liquor flow rate of 7490 L/min. The percent flow differences in each case were compared with the percent K/S differences measured on the corresponding sections of the dyed fabric beam.

The effect of yarn package density profile on the liquor flow profile was determined for the density profiles HML, MHL, HHL, uniform, LHH, and LMH at a constant overall package density of 500 g/L. With an HML density profile, “H” means high density, “M” means medium density, and “L” means low density. In addition, the flow rate was held constant at 80 gal/min, and to simulate leakage at both end-caps, the permeability constant at the end-caps was set at $1.00 \times 10^{-13} \text{ m}^2$. Although industry usually winds yarn packages to a uniform density for dyeing, this research studied how variable density within packages affects dye liquor flow because previous research indicated that variable density within packages can give more level dyeing than can a uniform package density (Yang and Mattison 1997).

The effect of collar placement on the liquor flow profile was studied with the collars placed under the fabric a distance based on a percentage of the beam winding height. For example, each beam had a winding height of 12.5 cm, and at a collar placement of 100%, the collars would cover 12.5 cm of perforations under the fabric on each side of the beam. Collar placements of 0%, 28%, 46%, 64%, 83%, 101%, 119%, 138%, 156%, and 174% were studied. Collars are normally used to cover the perforations in the metal beam not covered by the fabric so that liquor only flows through the fabric.

The effect on the liquor flow profile of placing end caps against the sides of the fabric beam was studied. Although end-caps are not used in beam dyeing, they are used in package dyeing to reduce axial flow and promote radial flow. For the 0% leakage condition, the end caps made the sides of the fabric beam impermeable to liquor flow. To model 75% leakage at the end caps, which is more realistic, a 1 cm space between the fabric and end caps was created, and the radial permeability constant of the 1 cm space was set a factor of 4 greater than that of the fabric beam. Another leakage condition was modeled in which the 1 cm space was modeled as being infinitely more permeable than the fabric beam.

The effect on the liquor flow profile of wrapping a low density fabric with a low permeability material was determined. The outer beam wrap was modeled as a low permeability material wrapped around the outer layer of the fabric beam by setting the radial permeability constant of the material to $3.98 \times 10^{-12} \text{ m}^2$. This effect was tested on a high permeability fabric with radial and axial permeability constants of $8.77 \times 10^{-9} \text{ m}^2$.

RESULTS AND DISCUSSION

Increasing the fabric beam density by winding the beam tighter from 550 g/L to 637 g/L reduced the percent flow differences for the side-center and the top-bottom due to reduced axial flow and decreased the percent K/S differences for those sections of the beam as shown in Figure 1. This result indicates that as liquor flow uniformity improves, the dye levelness improves as well. Winding the fabric beam tighter promotes flow in the radial direction because it increases the ratio of the radial to axial permeability constants. As a fabric beam is wound tighter, the permeability in the axial direction decreases at a faster rate than in the radial direction.

Winding yarn packages tightly at the inner layers and more loosely at the outer layers of yarn improved the liquor flow uniformity and reduced axial flow throughout the yarn packages. As shown in Figure 2, the HML, MHL, and HHL yarn package density profiles resulted in lower percent axial flow compared to a uniform package density profile, which indicates that a greater proportion of liquor tended to flow in the radial direction through the outer layers of yarn. The outer zones of the yarn packages were more permeable than the inner zones, and therefore the liquor was less likely to short circuit out the sides of the yarn packages near the dye tube. With the reduced axial flow, the percent flow differences decreased as well. In contrast to the HML, MHL, and HHL density profiles, the LHH and LMH density profiles resulted in a greater percentage of axial flow compared to a uniform density profile, which indicated that the liquor was more likely to bypass the outer zones and leak out the sides of the yarn packages near the dye tubes. These density profiles also gave higher percent flow differences.

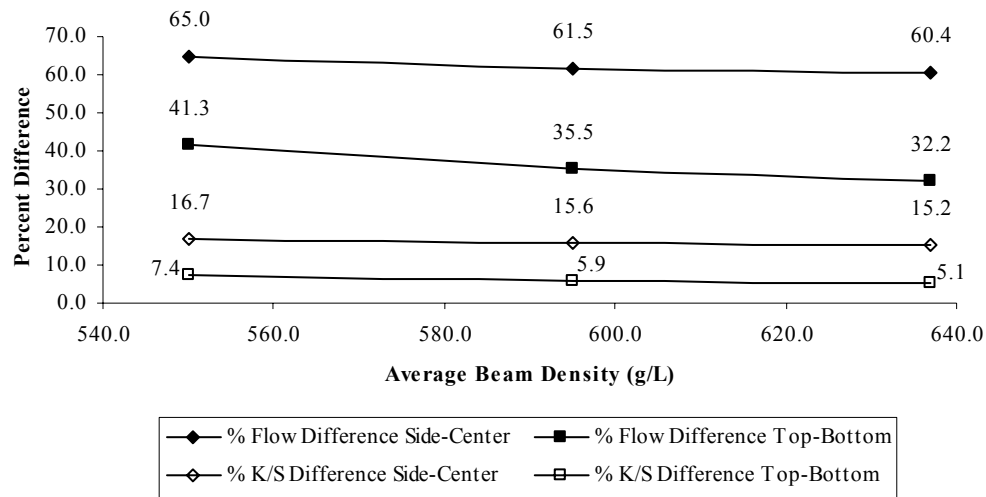


Figure 1: Percent Difference in Linear Velocity of Liquor and Percent Difference in K/S Values between the Sides and Center in the Outer Fabric Layers (Side-Center) and between the Outer and Inner Fabric Layers at the Center of the Beam (Top-Bottom) for Various Beam Densities at a Constant Flow Rate of 7490 L/min

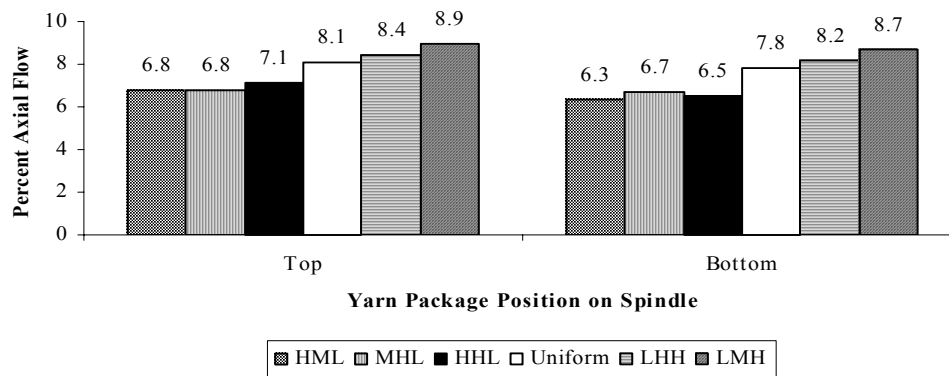


Figure 2: Percentage of Dye Liquor Flow in the Axial Direction for Various Package Density Distributions for the Packages at the Top and Bottom Position on the Spindle during Dyeing at Constant Flow Rate (80 gal/min), Overall Package Density (500 g/L), and Permeability Constant at the End-Caps ($1.00 \times 10^{-13} \text{ m}^2$)

For fabric beam dyeing, as the collar placement was decreased from 174% to 28% of the fabric winding height at a constant differential pressure of 2.15 bar, the side-center percent flow differences decreased as shown in Figure 3. Placing the collars under the fabric but closer to the fabric edge reduced the proportion of liquor flow short-circuiting out the sides of the fabric beam and promoted liquor flow at the top of the beam near the fabric edge. With the collars placed further under the fabric, liquor flow at the top side of the fabric beam was reduced because a greater proportion of liquor tended to flow axially out the sides of the beam near the inner layers of fabric. With a collar placement of 0%, which is the same as placing the collars over the perforations in the beam but not under the fabric, the side-center percent flow difference was greater than with a collar placement of 28%, and their top-bottom percent flow differences were about equal. Thus, the optimum collar placement is about 30% of the winding height because it minimizes the percent flow differences.

End-caps that acted as a barrier to axial flow during fabric beam dyeing forced the liquor to flow only in the radial

direction, and the side-center percent flow difference approached zero when 0% leakage was assumed as shown in Figure 4. The absence of axial flow minimized the percent flow difference between the top-bottom to 26%. For 75% leakage at the end-caps, the resulting percent flow differences for the side-center and top-bottom were still lower compared to the use of collars. Thus, the use of end-caps in fabric beam dyeing would be expected to improve dye levelness.

A low density fabric gave an asymmetrical liquor flow profile that was affected by the suction towards the drain as shown in Figure 5. Because the fabric offered very low flow resistance, eddies formed inside the metal beam, and as the liquor entered the fabric, the liquor tended to shortcut through the fabric beam and flow directly to the drain. Wrapping a high permeability fabric beam with a low permeability material improved the liquor flow profile by directing flow through the fabric beam and eliminating regions of low flow.

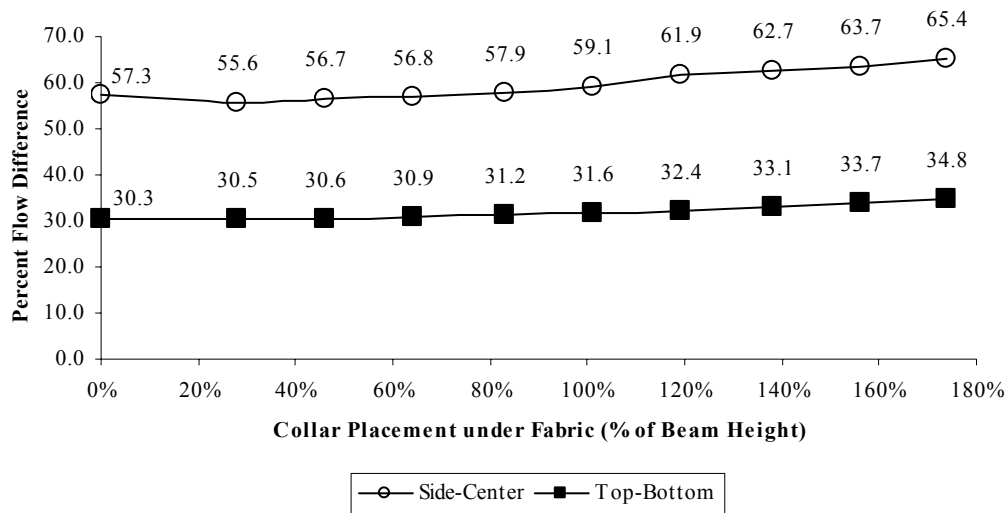


Figure 3: Effect of Various Collar Placements on the Percent Difference in Linear Velocity of Liquor between the Sides and Center of the Beam in the Outer Fabric Layers (Side-Center) and between the Outer and Inner Fabric Layers at the Center of the Beam (Top-Bottom) at a Constant Differential Pressure of 2.15 bar

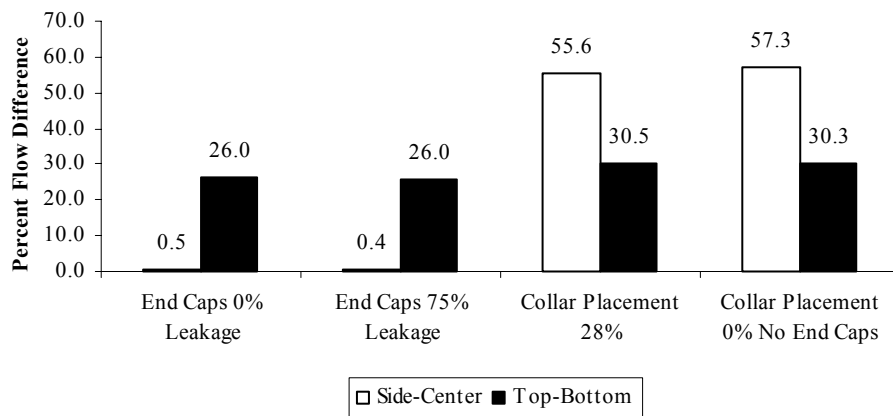


Figure 4: Effect of End Caps or Collars on the Percent Difference in Linear Velocity of Liquor between the Sides and Center of the Beam in the Outer Fabric Layers (Side-Center) and between the Outer and Inner Fabric Layers at the Center of the Beam (Top-Bottom) at a Constant Differential Pressure of 2.15 bar

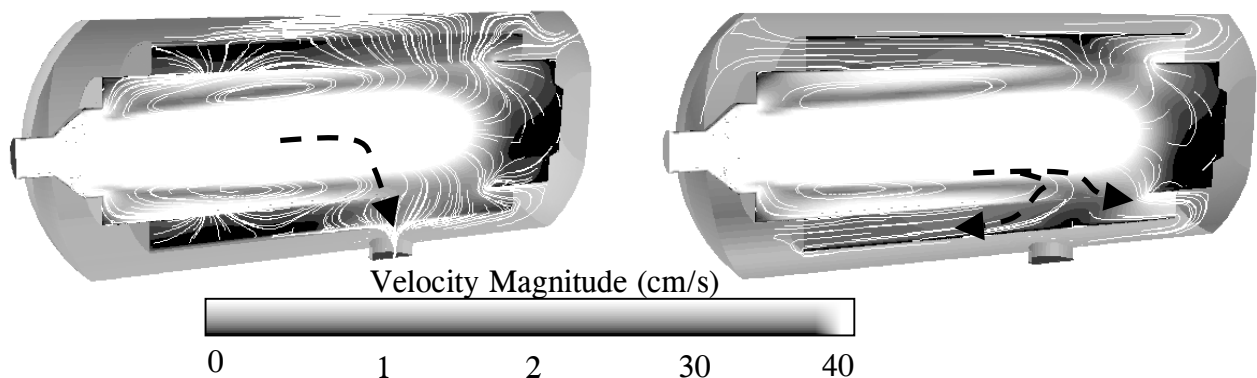


Figure 5: The Path of Dye Liquor through a Low Density Fabric Beam without Outer Beam Wrap (Left) and with Outer Beam Wrap (Right) with White Lines and Black Dotted Lines Indicating Path of Flow

CONCLUSIONS

CFD was used to model dye liquor flow during fabric beam dyeing and yarn package dyeing. Decreasing percent flow differences calculated with CFD corresponded to

decreasing percent K/S differences measured on fabric dyed under the same conditions, which indicated that this was a valid model of liquor flow in fabric beam dyeing and yarn package dyeing. The results suggest that as liquor flow uniformity improves, the levelness of dyeing

improves. The liquor flow uniformity can be improved by promoting radial flow through the packages and beam and by reducing axial flow. This can be accomplished by winding the fabric beam more tightly overall or by winding yarn packages or fabric beams more tightly at the inner layers and more loosely at the outer layers of yarn or fabric. To further improve liquor flow uniformity, collars can be placed under the fabric a distance of about 30% of the winding height on each side of the fabric beam, or end caps can be placed against each side of the beam to almost eliminate axial flow. For low density fabric, the liquor may tend to flow directly to the drain rather than through the entire fabric beam. To improve flow uniformity in this case, the fabric beam can be wrapped with a low permeability material, and liquor flow will be directed away from the drain and through the entire fabric beam.

molecular modeling software to model the interactions between dye molecules and polymer molecules.

ACKNOWLEDGEMENTS

The authors are grateful to Gilbert O'Neal, Henry Boyter, Greg Hughey and Dennis Balmforth for their constructional comments on the research. Financial supports from Milliken & Company, Institute of Textile Technology and University of Nebraska-Lincoln, Agricultural Research Division are also gratefully acknowledged.

REFERENCES

- Farber, P., Dahmen, N., and Mohaupt, H. 1999. "Virtual Prototyping in Textile Finishing: Academic Pastime or Industrial Reality". *Melliand International* 1, No. 3 (Mar), 75-78.
- Fluent, Inc. 2000. *Fluent Software Training*. Available from Fluent, Inc., 10 Cavendish Court, Lebanon, New Hampshire 03766, USA.
- Gibson, P. 1999. "Water Vapor Transport and Gas Flow Properties of Textiles, Polymer Membranes, and Fabric Laminates". *Journal of Coated Fabrics* 28, No. 4 (Apr), 300-327.
- Kulichenko, A., and Langenhove, L. 1992. "Resistance to Flow Transmission of Porous Materials". *Journal of the Textile Institute* 83, No. 1 (Jan), 127-132.
- Rapp, W., Yang, Y., O'Neal, G., Boyter, H. and Hughey, G. 2002. "Evaluating Liquor Flow Profiles in Fabric Beam Dyeing". *AATCC Review* 2, No.2 (Feb), 42-46.
- Yang, Y., and Mattison, V. L. 1997. "The Effect of Package Density Profile on Levelness of Package Dyed Yarn". *Textile Chemist and Colorist* 29, No. 8 (Aug), 77-81.

AUTHOR BIOGRAPHY

DAVID KARST was born in Silverspring, Maryland, USA and attended Virginia Tech in Blacksburg, Virginia, USA, where he obtained a B.S. in Chemical Engineering. He then enrolled in the Institute of Textile Technology in Charlottesville, Virginia, USA, and earned an M.S. in Textile Technology. For his thesis, he used Computational Fluid Dynamics to model dye liquor in yarn package dyeing. He is currently working towards a Ph.D. in Textile Science at the University of Nebraska in Lincoln, Nebraska, USA, and his dissertation will involve using

DYEING PROCESS MODELING FOR POLYESTER MICRO FIBERS

Keun - Hoo Park,

Vladan Koncar

Génie et Matériaux Textiles (GEMTEX) Laboratory,

Ecole Nationale Supérieure des Arts et Industries Textiles

9 rue de l'Ermitage - BP 30329 - 59056 ROUBAIX, FRANCE

E-mail: vladan.koncar@ensait.fr

KEYWORDS

Micro fibers, Modeling, Polyester

ABSTRACT

This paper deals with the mathematical modeling of the diffusion during a dyeing process of the polyester micro fibers and super micro fibers. This model applies to the dyeing of different polyester fibers with disperse dyes. The influence of temperature variations, the initial dye concentration and the fiber count on the diffusion coefficient is the main subject of this study. Simulations of dyeing process are carried out using the diffusion coefficient model in order to compute the concentration of dyes into the micro fibers in function of the time and other dyeing parameters.

INTRODUCTION

Many papers have dealt with the kinetics of dyeing but the problem is to know whether the equations that have been proposed represent the real physical mechanism (Koncar and Casetta 2001). Generally, the dye diffusion into a fiber is described mathematically by Hill and Wilson's equations (or their approximations) which compare the diffusion into a textile fiber with the diffusion into a cylinder of infinite length (Johnson and al. 1989; Vickerstaff and al. 1954; Shibusawa 1979; Etters and al. 1980;...) This approach is based on the second Fick's law. Moreover, these models consider that the diffusion coefficient is constant during the entire dyeing process and thus neglect the dye concentration and the dyebath temperature influence on the diffusion. This article is two-fold: first the temperature variations and the fiber count influences on the diffusion coefficient value is analyzed and modeled. Finally, we have validated this model by making an experiment with a given initial dye concentration and a temperature profile that implies the diffusion coefficient variation. The dyeing uptake rate results coming from the previous experiment are compared to simulation results obtained from our diffusion model and dyeing kinetics model.

COMPUTATION OF THE DIFFUSION COEFFICIENT D

Shibusawa's approximations of Hill or Wilson's equations (Shibusawa 1985) are applied in order to compute the diffusion coefficient value at each sampling time from the dyeing uptake rate M_t/M_∞ obtained from experimental dyeing.

Etters' method

The procedure used to compute the diffusion coefficient D in our previous paper (Koncar and Casetta 2001) has been based on Etters' method. At the point of transition from an infinite to a finite dyebath, Etters distinguishes two forms of fractional diffusion uptake (Etters 1994):

$f_{sat} = \frac{M_t}{M_\infty}$: fractional saturation uptake for infinite bath

conditions *i.e* for M_t values ranging from 0 to $(M_t)_{transition}$.

$f_\infty = \frac{M_t}{M_\infty}$: fractional equilibrium uptake for finite bath

conditions *i.e* for M_t values ranging from $(M_t)_{transition}$ to M_∞ .

These two quantities are related to each other by:

$$f_{sat} = \frac{\alpha f_\infty}{1 + \alpha - f_\infty} \quad (1)$$

where α is given by: $\alpha = \frac{1 - E_\infty}{E_\infty} = \frac{Q}{K_\infty}$

where: Q is the liquor ratio and K_∞ is the partition coefficient of the dye/fiber system

$E_\infty = \frac{C_0 - C_\infty}{C_0}$ is the fractional equilibrium exhaustion

C_0 is the initial dye concentration

C_∞ is the equilibrium concentration of dye in the dyebath

According to Mc Gregor (Mc Gregor and Etters 1979), the transition from an infinite to a finite bath occurs when:

$$f_\infty = f_{sat} \left[\frac{C_{sat}}{\alpha C_0 E_\infty} \right] \quad (2)$$

where C_{sat} is the aqueous saturation solubility of the dye.

Etters' method (Etters 1994) was used to determine the diffusion coefficient evolution as a function of time from experimental values. From the beginning of the dyeing experiment to the transition moment, f_{sat} is determined for each sampling time. Then, the Shibusawa's approximation of Hill's equation is used to compute the corresponding diffusion coefficient. As for the finite bath part, going from the transition moment to the end of the dyeing, f_∞ is determined, and thanks to Shibusawa's approximation of Wilson's equation, the diffusion coefficient is computed. Finally, the diffusion coefficient evolution as a function of time is obtained.

Casetta-Koncar method

This method, used to obtain the D evolution in the case of a transitional dyebath, is as follows:

a/ Condition for an infinite dyebath:

$$\left(\frac{M_t}{M_\infty} \right)_{t=0} < \left(\frac{M_t}{M_\infty} \right)_t < \left(\frac{M_t}{M_\infty} \right)_{transition} \quad i.e: 0 < \left(\frac{M_t}{M_\infty} \right)_t < \left(\frac{M_t}{M_\infty} \right)_{transition}$$

b/ Condition for a finite dyebath:

$$\left(\frac{M_t}{M_\infty}\right)_{transition} < \left(\frac{M_t}{M_\infty}\right)_t < \left(\frac{M_t}{M_\infty}\right)_{t=\infty} \quad i.e:$$

$$\left(\frac{M_t}{M_\infty}\right)_{transition} < \left(\frac{M_t}{M_\infty}\right)_t < 1$$

where $\left(\frac{M_t}{M_\infty}\right)_t$ is the measured dyeing uptake rate obtained

from transmittance values (spectrophotometer).

Hill and Wilson have considered infinite and finite dyebaths separately. They supposed that at the end of the dyeing process, the equilibrium is reached and that during the whole process the bath type does not change (infinite or finite). This means that when Hill's or Wilson's equation is used, the $\left(\frac{M_t}{M_\infty}\right)$ ratio varies from 0 to 1.

Nevertheless, in the case of a transitional dyebath, the equilibrium is not reached at the transition moment (infinite to finite bath transition) and $\left(\frac{M_t}{M_\infty}\right)$ value is different from 1

at the transition point. To apply Hill and Wilson's equations in that case, the two systems (infinite and finite baths) must be considered separately.

Therefore, for each type of bath, the $\left(\frac{M_t}{M_\infty}\right)_t$ ratio has to vary

from 0 to 1. The following calculations must be done so that the previous condition is fulfilled:

In the case of an infinite dyebath:

$$\left[\left(\frac{M_t}{M_\infty}\right)\right]_{infinite} = \frac{\left(\frac{M_t}{M_\infty}\right)_t}{\left(\frac{M_t}{M_\infty}\right)_{transition}} \quad \text{for:}$$

$$0 < \left(\frac{M_t}{M_\infty}\right)_t < \left(\frac{M_t}{M_\infty}\right)_{transition}$$

in order to modify the upper limit value of $\frac{M_t}{M_\infty}$ from

$$\left(\frac{M_t}{M_\infty}\right)_{transition} \quad \text{to } 1.$$

Then, Shibusawa's approximation of Hill's equation is applied for: $0 < \left[\left(\frac{M_t}{M_\infty}\right)\right]_{infinite} < 1$ in order to obtain the

diffusion coefficient.

In the case of a finite dyebath:

$$\left[\left(\frac{M_t}{M_\infty}\right)\right]_{finite} = \frac{\left(\frac{M_t}{M_\infty}\right)_t - \left(\frac{M_t}{M_\infty}\right)_{transition}}{1 - \left(\frac{M_t}{M_\infty}\right)_{transition}} \quad \text{for:}$$

$$\left(\frac{M_t}{M_\infty}\right)_{transition} < \left(\frac{M_t}{M_\infty}\right)_t < 1$$

in order to modify the lower $\left(\frac{M_t}{M_\infty}\right)$ value from $\left(\frac{M_t}{M_\infty}\right)_{transition}$

to 0.

Then, Shibusawa's approximation of Wilson's equation is applied for: $0 < \left[\left(\frac{M_t}{M_\infty}\right)\right]_{finite} < 1$

The diffusion coefficient value is computed thanks to $\left[\left(\frac{M_t}{M_\infty}\right)\right]_{infinite}$ and $\left[\left(\frac{M_t}{M_\infty}\right)\right]_{finite}$ values, and the approximations

of Hill and Wilson's equations. Finally, the diffusion coefficient evolution is obtained as a function of time and the conditions of validity of Hill and Wilson's equations are respected.

INFLUENCES OF TEMPERATURE VARIATIONS AND FIBER COUNT ON D

A model describing the influence of initial dye concentration and time on the diffusion process has been developed recently (Koncar and Casetta 2001). However, this model cannot be applied to commercial dyeing systems because the dyeing process is not an isothermal one. Instead of using the average value of D in order to compute the dyeing uptake rate, we know that temperature influences the diffusion process. On the other side the fiber count has also an important influence on the diffusion coefficient. In fact, the fibre surface area increases strongly as the fibre count decreases. The crystallinity index maintains stable for all the fibres.

Therefore, the consequences are that micro fibres have higher dyeing capacity of than of conventional fibres. The dye molecules are fixed on available dyeing sites. For higher dye concentrations, there are more molecules to be fixed on available dyeing sites.

The aim of this paper is to quantify these temperature and fiber count influences on the diffusion coefficient.

Importance of temperature on dyeing results

A number of attempts have been made to choose dye exhaustion profiles for the optimal control of dyeing process. All these approaches have been based on the theoretical isothermal dye-uptake curve, although the dyeing process is not an isothermal one (Popescu and Segal 1984). A typical plot of temperature against time of dyeing is shown in Fig. 1, from which it is obvious that isothermal kinetics apply only to stages I and III. During stages II and IV, the dyebath temperature varies. The second stage is recognized as being of great importance for correct dye exhaustion and good leveling of the dyeing. These parameters determine the environmental impact and the quality of dyeing.

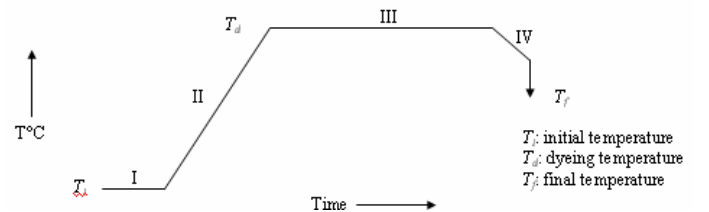


Figure 1: Typical profile of dyeing temperature against time

Owing to the low rate of diffusion of disperse dyes into polyester fibers at temperatures up to the boil (~100°C), the use of high temperatures (125-130°C) enables the dyeing rate to be increased to a commercially acceptable level

(Burkinshaw 1995). For instance, Rochas (Rochas 1960) demonstrated that dyeing of equal depth could be obtained after 11 seconds at 200°C, 5 minutes at 125°C and 5 days at 60°C. Merian (Merian and al. 1963) found that the rate of dyeing at 130°C was some 200 times greater than at 100°C, showing that dyeing rate increases about six-fold for a temperature increase of 10°C.

The greater rate of dyeing and generally superior dye migration obtained for high dyeing temperatures can be considered to be a consequence of the higher kinetic energy of the dye molecules, the greater segmental mobility of the less-ordered regions within the fiber and the higher aqueous solubility of the dye at elevated temperatures. In fact, the temperature at which dyeing must be carried out is above the glass transition temperature (T_g) of the polymer in order that polymer segments motion becomes possible. Below T_g , segmental motion of the main chain has ceased with a corresponding increase in the resistance to the penetration of dye molecules. Dyeing rates have, in fact, been shown to be negligible below T_g ; once the temperature reaches this value, major increases occur (Peters 1975). The dye diffusion into the fiber for temperatures superior to the glass transition temperature (T_g) can be expressed by the following expression [18]:

$$\log \frac{D_T}{D_{T_g}} = \frac{A(T - T_g)}{B + (T - T_g)} \quad (3)$$

where T is the dyebath temperature, T_g the glass transition temperature, D_T the diffusion coefficient value at temperature T , D_{T_g} the diffusion coefficient value at T_g , and A and B constant experimental parameters. A large number of polymers have been found to conform to equation (3) over the temperature range from T_g to $T_g + 100$ (°C) (Williams and al. 1955).

Thus, equation (3) indicates the temperature influence on the diffusion coefficient. On the other side, this equation does not include the initial dye concentration, the fiber count and the time influence on D as developed in our diffusion model.

Experimental results for different dyeing temperatures (for the same initial dye concentration)

The diffusion model obtained in our previous study has been developed in order to include the influence of temperature variations on the diffusion coefficient value. The procedure used to obtain the apparent diffusion coefficient has been described previously. D is modeled according to the dyebath nature. Table 1 summarizes the dyebath nature in function of the dyeing temperature for 4% initial dye concentration (an example), the transition values of $\frac{M_t}{M_\infty}$ and the

mathematical models used to describe the diffusion coefficient kinetics.

Importance of fiber count on dyeing results

A microfiber is traditionally defined as a fiber or filament of linear density below approximately 1 dtex (i.e. fiber count). However, even finer fibers of less than 0.3 dtex are produced, these commonly being referred to as supermicrofibers or ultra fine fibers, (the dtex is the mass in grams of 10 km of fibre).

Table 1: Nature of dyebath, 4% initial dye concentration

Dyebath Temp.	Dyebath nature	$\left(\frac{M_t}{M_\infty}\right)_{transit}$	Mathematical model
135°C (4.14%)	Transitional bath	0.8191	$D_{inf-bath}, 0 < M_t/M_\infty \leq 0.8191$ $D_{f-bath}, 0.8191 < M_t/M_\infty < 1$
130°C (4.22%)	Transitional bath	0.8637	$D_{inf-bath}, 0 < M_t/M_\infty \leq 0.8637$ $D_{f-bath}, 0.8637 < M_t/M_\infty < 1$
125°C (3.94%)	Transitional bath	0.8900	$D_{inf-bath}, 0 < M_t/M_\infty \leq 0.8900$ $D_{f-bath}, 0.8900 < M_t/M_\infty < 1$
120°C (4.03%)	Transitional bath	0.9116	$D_{inf-bath}, 0 < M_t/M_\infty \leq 0.9116$ $D_{f-bath}, 0.9116 < M_t/M_\infty < 1$
115°C (4.14%)	Transitional bath	0.9295	$D_{inf-bath}, 0 < M_t/M_\infty \leq 0.9295$ $D_{f-bath}, 0.9295 < M_t/M_\infty < 1$

It is obvious that the physical, mechanical and specially dyeing properties of these different microfibers are quite different. First, the reduction of filament linear density is also accompanied by an increase in the surface area per unit volume of the filament. In fact, the specific surface increases markedly with decreasing filament linear density.

The polyester multifilament yarns, 0.07 dtex (Sea-Islands fiber type) 0.22 dtex (77dtex/300fil), 0.56 dtex (83dtex/144fil) and 1.46 dtex (292dtex/200fil) are used in our experiments. All these fibers were obtained from Hyusung Inc., South Korea.

Therefore, we have studied the influence of the fiber count on the diffusion coefficient model parameters. The results of this study are shown in the next sections.

3.4 Mathematical model of the diffusion coefficient

The mathematical model defined from experimental curves giving the diffusion coefficient evolution against time in function of dyeing parameters is given by the expression (4).

Infinite bath

$$D_{inf_bath}(t, C_0, FC, T) = a_i t^{b_i} \exp(-c_i t^{d_i}) \quad (4)$$

where:

t : time (s)

C_0 : initial dye concentration (g/L)

D : diffusion coefficient (cm²/s)

FC : fiber count (dtex)

T : temperature (°K)

The diffusion coefficients for microfibers and conventional fibres are analyzed only during the infinite dyebath phase for infinite and transitional dyebaths. The diffusion coefficient for finite dyebath configuration corresponding to very low initial dye concentrations has not been studied. In fact the D values obtained in the case of finite dyebath condition are much lower than that obtained previously (infinite dyebath).

Modeling and identification

The diffusion coefficient values obtained from the model and those computed from experimental $\frac{M_t}{M_\infty}$ values and

Shibusawa's approximations of Wilson and Hill's equations are compared.

Then, the genetic algorithms based optimization program is used to identify all the parameters. The same procedures as described previously (Koncar and Casetta 2001) are used. This optimization method is applied for all combination of initial dye concentration, dyeing temperature and fiber counts.

The identification results show that despite the change in the calculation method, c_i parameter is still independent of the initial dye concentration if the temperature is unvarying (tendency similar to that obtained with Etters' method). Moreover, for a given initial dye concentration, b_i and d_i parameters do not vary whatever the temperature of application.

Equations describing the evolution of each parameter as a function of the initial dye concentration C_0 , the temperature T and the fiber count FC are obtained by minimizing the sum of quadratic errors between the identified parameter equation values and the same parameter table values. All the results are given in the Table 2.

Table 2: Model parameters

a_i : log fit	b_i : power fit
$a_i = \varphi * \ln(FC \text{ (dtex)}) + \psi$ $\varphi = 5E-12 * T(^{\circ}K) - 2E-9$ $\psi = 2E-12 * T(^{\circ}K) - 6E-10$	$b_i = 0.0292 C_0^{-0.9266}$
c_i : linear fit	d_i : power fit
$c_i = 0.0017 T(^{\circ}K) - 0.629$	$d_i = 0.4349 C_0^{-0.0678}$

Discussion

The equation (4) describing the diffusion coefficient evolution according for infinite dyebath has been obtained, containing four parameters. b_i and d_i parameters are independent of the dyeing temperature whatever the initial dye concentration. Moreover, the results obtained after the optimization step show that the parameter c_i depends only on the temperature. Parameter a_i is more difficult to model. In fact, it depends strongly on the fiber count; however the dyebath temperature modifies slightly this dependence.

DYEING PROCESS GLOBAL MODEL

Structure

The main objective of our study is to develop a global model including both the dye diffusion and the dye concentration evolution into polyester. The diffusion is described by the diffusion coefficient $D(t, C_0, T)$ and the dye concentration into polyester fibers by the $\frac{M_t}{M_{\infty}}$ ratio where M_t is the dye

concentration into the fiber at time t and M_{∞} the dye concentration at the equilibrium state.

In its final configuration, the proposed model is defined in the form of a non-linear state space model (5).

$$\dot{X}(t) = f(X(t), U(t), t) \quad \text{state equation} \quad (5)$$

$$S(t) = g(X(t)) \quad \text{output equation}$$

($S(t)$ is the color of the textile product at time t)

$$\text{where: } X(t) = \begin{bmatrix} \frac{M_t}{M_{\infty}} \\ D(t, C_0, T) \end{bmatrix} \quad \text{system state}$$

$$U(t) = \begin{bmatrix} T \\ C_0 \end{bmatrix} \quad \text{system input}$$

The exogenous parameters relative to the dyeing process are polyester properties, dyes properties and the dyeing machine properties. These parameters influence dyeing models parameters equations that have to be identified for each set of exogenous parameters.

Equations analysis

A fine analysis of the equations is given in order to highlight the complexity of the observed phenomena.

The nature of the dyeing process (batch under high pressure), the $S(t)$ output measured with the spectrophotometer (measurement by reflection) is accessible to measurement only at the end of the dyeing, on the dried products (yarn bobbin or fabric). The diffusion coefficient $D(t, C_0, T)$ cannot be measured directly. The output equation model, $g(\cdot)$, is not the object of this article.

The objective of modeling is to establish the direct relation between the product's final color and the measurable parameters during the dyeing like the dyebath temperature and the initial dye concentration and indirectly throw the model parameters involving exogenous parameters.

The functions $f(\cdot)$ and $g(\cdot)$ thus characterize at the same time the dye/polyester system and the dyeing machine properties by including the physicochemical phenomena knowledge.

Therefore, the global model of the dyeing process can be expressed as follows (equation (6)).

$$\begin{bmatrix} \frac{d\left(\frac{M_t}{M_{\infty}}\right)}{dt} \\ \frac{dD(t, C_0, T)}{dt} \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{\infty} \frac{4}{r^2} [D + t.D] \exp\left(-\frac{b_n D(t, C_0, T).t}{r^2}\right) \\ \left(\frac{b_t}{t} - c_i.d_i.t^{d_i-1}\right) D(t, C_0, T) \end{bmatrix} \quad (6)$$

The dyebath temperature and the initial dye concentration appear through the parameters b_i , c_i and d_i . The first and second expressions of equation (6) are the differential equations whose solutions are respectively Hill's model and our diffusion model. In that second case, the initial condition is as follows: $D_{(t=0)} = a_i$.

To simplify the presentation, we only developed in this demonstration the global dyeing model in the case of the infinite bath with a sufficient flow rate (Hill's equation). In the case of the finite bath and/or insufficient flow rate, the global model varies in connection with the approaches of Wilson or also Crank or Newman for insufficient flow rates.

APPLICATION TO A DYEING EXPERIMENT

Finally, a dyeing experiment is realized for a fixed initial concentration and for a dyebath temperature profile generally used in real dyeing conditions.

CONCLUSION

The modeling procedure resulted in the equation describing the diffusion coefficient evolution as a function of time, initial dye concentration, fiber count and temperature.

This model contains four parameters a_i , b_i , c_i and d_i : b_i and d_i depend only on the initial dye concentration c_i varies according to the temperature and a_i depends on the fiber count and on the dyebath temperature.

REFERENCES

- V. Koncar & M. Casetta, Dynamic modelling of the diffusion in a dyeing process, *International Journal of Modelling and Simulation*, Vol.21, N°3, 2001, 191-200.
- A. Johnson, *The Theory of Coloration of Textiles* (Second Edition, Society of Dyers and Colourists, 1989).
- T. Vickerstaff, *The Physical Chemistry of Dyeing* (Second edition) (Published for Imperial chemical industries limited by Oliver and Boyd, 1954).
- T. Shibusawa, Approximations Describing the Rate of Diffusion of Dye into an Infinite Cylinder and Plane Sheet from a Constant surface Concentration, *JSDC*, 95(5), 1979, 175-177.
- J. N. Etters, Diffusion equations made easy, *Text. Chem. Col.*, 12(6), 1980, 140-145.
- A. Urbanik, Further Equations for Approximating the Rate of Dyeing Curve in Diffusion-Controlled Processes, *JSDC*, 92, 1976, 377-379.
- W. F. Kilby, The Relation of the Apparent Diffusion Coefficient to the Time of Half-Dyeing, *JSDC*, 76, 1960, 479-484.
- Tsuda, Bulletin of the Text. Research Inst., Japan, 36, 1962, 51.
- Inoue et Suda, *Sen-i Gakkaishi*, 24, 1968, 88.
- A. Urbanik, Equation for Diffusion-Controlled Dyeing from a Finite Bath Simplified Rate of Dyeing, *JSDC*, 90, 1974, 444-445.
- T. Shibusawa, Wide Ranging Polynomial Approximations to Wilson's and Hill's Equations, *JSDC*, 101(7/8), 1985, 231-234.
- J.N. Etters, Sorption of Disperse Dye by Polyester Fibers: Boundary Layer Transitional Kinetics, *Textile Research Journal*, 64(7), 1994, 406-413.
- R. McGregor and J.N. Etters, Transitional Kinetics in Disperse Dyeing, *Textile Chemist and Colorist*, 11, 1979, 202-206.
- C. Popescu & E. Segal, Non-isothermal Kinetic Method in the Study of Dyeing Processes, *JSDC*, 100(12), 1984, 399-400.
- S. M. Burkinshaw, *Chemical Principles of Synthetic Fibre Dyeing* (Blackie Academic & Professional, Chapman & Hall, 1995).
- P. Rochas, *Chemiefarsen*, 10, 1960, 666.
- E. Merian, Carbonell J., Lerch U. et Sanahuja V., The Saturation Values, Rates of Dyeing, Rates of Diffusion and Migration of Disperse Dyes on Heat-set Polyester Fibres, *JSDC*, 79(11), 1963, 505-515.
- R.H. Peters, *Textile Chemistry*, Volume III, The physical Chemistry of Dyeing, (Amsterdam: Elsevier, 1975).
- M.L. Williams, R.F. Landel & J.D. Ferry, The Temperature Dependence of Relaxation Mechanisms in Amorphous Polymers and Other Glass-forming Liquids, *J. Amer. Chem. Soc.*, 77(7), 1955, 3701-3706.

BIOGRAPHY

KEUN HOO PARK PhD student in advanced analyses and instrumentation from University of Lille 1 (France) since December 2001.

Research laboratory: GEMEX (Génie et Matériaux textiles) at the Ecole Nationale Supérieure des Arts et Industries Textiles (ENSAIT) of Roubaix (France). Title of thesis: Diffusion of

disperse dyes into micro fibres and conventional polyester fibers.

VLADAN KONCAR was born in 1962 and went to the University of Lille 1, where he did his PhD in 1991. He is currently Professor at ENSAIT (Engineering Institute for Textile). His research fields of interest are Predictive control, Compound control, Robust control of complex systems, Modeling of dyeing processes, Nonlinear optimization, Virtual reality and communicative clothing.

APPENDIX

Hill and Wilson's equations

The uptake of disperse dyes by cylindrical fibres is described using the diffusion equations solutions of Hill and Wilson, in the case of a sufficient flow rate of the dyeing machine.

To describe the isothermal kinetics of dyeing of PET fibres from a finite dyebath, Wilson's equation (7) is used:

$$\frac{M_t}{M_\infty} = 1 - \sum_{n=1}^{\infty} \frac{4a(1+a)\exp(-q_n^2(D_f t / r^2))}{4 + 4a + a^2 q_n^2} \quad (7)$$

where: M_t = concentration of dye on the fibre at a given time t

M_∞ = concentration at equilibrium

D_f = diffusion coefficient of the dye in the fibre (cm²/s)

t = dyeing time (s)

r = fibre radius (cm)

q_n = positive, non zero roots of the equation: $\alpha q_n J_0(q_n) + 2J_1(q_n) = 0$

J_0 and J_1 = zero- and first-order Bessel functions.

a is given by: $a = \frac{1 - E_\infty}{E_\infty}$ where E_∞ is the fractional

equilibrium exhaustion

(7) is reduced to Hill's equation (8) when fractional exhaustion $E_\infty = 0$ i.e. to describe the rate of dyeing from an infinite dyebath:

$$\frac{M_t}{M_\infty} = 1 - \sum_{n=1}^{\infty} \frac{4}{b_n^2} \exp(-b_n^2(D_f t / r^2)) \quad (8)$$

where: b_n = positive, non zero roots of the equation: $J_0(b_n) = 0$, and J_0 = zero-order Bessel function

Shibusawa's approximations of Hill and Wilson's equations

Approximation of Hill's equation

$$\frac{M_t}{M_\infty} \approx 1 - \left(\frac{1 - 0.77199X}{1 + 0.352474X + 0.64404X^2 - 0.10277X^3 + 3.13748X^4} \right)^2$$

$$\text{where } X = \left(\frac{Dt}{r^2} \right)^{1/2} \quad (9)$$

$$\Delta(M_t/M_\infty) = \pm 0.000083 \text{ for } 0 < M_t/M_\infty < 0.998$$

Approximation of Wilson's equation

$$\frac{M_t}{M_\infty} \approx \frac{A_1 X + A_2 X^2}{1 + B_1 X + B_2 X^2}, \text{ where } X = \left(\frac{Dt}{r^2} \right)^{1/2} \quad (10)$$

USE OF DISCRETE SIMULATION TO DESIGN AND MANAGE AN INDUSTRIAL DYEING LABORATORY

Giorgio Romanin-Jacur
Department of Management and Engineering
University of Padova
Stradella S. Nicola 3
I 36100 Vicenza, Italy
E-mail: romjac@dei.unipd.it

KEYWORDS

Knitting yarn production, dyeing laboratory, resource allocation, management, simulation.

ABSTRACT

The dyeing laboratory of a factory producing knitting yarn is considered with the scope of increase its performances and consequently reduce the products time to market. After an accurate analysis of the laboratory in its current structure and organisation, a flexible general model is built up and implemented in Micro-Saint, in order to evaluate and compare the effects of a set of alternative changes in the structure and in the management.

INTRODUCTION

In the textile-apparel and footwear field, time to market is an important competition element, given the necessity of a continuous product updating. In a factory producing knitting yarn, dyeing is an essential part of production cycle, therefore any error or delay in the creation and testing of colour recipes and dyeing processes may cause dramatic damage. The dyeing laboratory is devoted to the creation and test of new recipes and related processes and to the correction of currently employed ones, therefore its performance is of great importance in the factory economy. In this paper modelling and simulation are employed to evaluate such a laboratory performance both in its current structure and in alternative suggested ones. A general flexible model, made up of a small number of connectable modules, has been studied and implemented for this scope.

ENVIRONMENT

The environment where our study was developed is the textile-apparel and footwear field, which is characterised by large market but high competition; a substantial competition element, able to determine the enterprise success, is constituted by the time to market, which requires to be shorter and shorter.

We consider a factory producing knitting yarn; yarn is sold to buyers for the production of a) underwear and neonatal knitwear, b) sweaters, shirts and similar, c) hosiery. Produced articles are about 60, of different material, both natural and synthetic; production is highly differentiated, in order to offer a large product spectrum, and requires to be frequently updated.

The need for a high quality production and a high level service towards customers in a strongly variable and unsure

market requires to increase work effectiveness and efficiency.

The considered factory is located in North-East Italy; about half of the production is sold in Italy and half abroad.

The factory includes three working centres, two of them devoted to yarn production and one devoted to yarn dyeing. In the dyeing centre yarn distaffs and skeins are mixed with colour, water and additives in pressurized pools, then yarn is softened, squeezed, dried and packed to be either stored or sent to the customer. A first quality control is effected in the pools after colouring; a second sample control is effected on finished products.

DYEING LABORATORY

The object of our interest is the dyeing laboratory, which constitutes the connection element and the bottleneck among yarn production and dyeing. Its macro-activity consists in creating "recipes" in order to correctly dye all articles; in other words the laboratory has to supply the right colour and to design the related dyeing process for all products. Recipes may concern:

- a) new articles, where the colour may be either suggested inside the factory by the design department (by supplying finished specific samples) or imposed by the customer (by supplying either paper, tissue, yarn samples or references to concurrent products); in the first case we are in presence of regular and planned requests with low urgency, in the second case the designed and tested recipe may either be required by the customer for a check (random request and high urgency) or directly sent to the production centre (random request but planned urgency);
- b) old articles for which the production was interrupted and has to be restarted, for which an old receipt and some specific samples are at disposition, but old components may no more be found or used; the same may happen consequent to periodical checks to currently used recipes which risk to "expire", i.e., to become old and need renewal (regular request and planned urgency);
- c) articles whose current production presents defects for what concerns colours (random request and very high urgency, as that stops the respective article production).

Every recipe is created by a "trial and error" process, where the error may be detected either inside the laboratory, by spectrophotometric analysis, or by the customer. Every recipe is designed by a specialized technician for what concerns colours, additives and dyeing process; the colour is

obtained either manually or by an automatic machine; then the recipe is tested by laboratory operators according to one of two different dyeing plans: "old", i.e., totally manual with simple machines, which is compulsory for particular yarn types, or "new", i.e., totally robotised, by a quick but complex machine which requires a longer preparation time; finally the resulting sample is carefully checked from the quality point of view; many samples may not satisfy the check, and in that case the related recipe need to be modified and the whole process to be repeated.

It is obvious that the entire laboratory activity needs to be accurately scheduled by the coordinator, according to above mentioned priorities, and all data related to tests and checks are stored in a computer according to precise codes.

LABORATORY POOR PERFORMANCE

At the beginning of the study the laboratory activity was characterised by 150-200 recipes supplied per week, which required 250-700 dyeing tests per week. Moreover the amount of weekly requests was not homogeneous during the year, as it presented two peak intervals in the Spring and in the Fall.

The laboratory was equipped with a manually ruled machine to obtain colours and 6 "manual" dyeing machines of old type, all requiring a high level of human involvement; it operated on two turns of eight hours each, supplied by seven operators. It is evident that the high amount of work required a careful scheduling in order to try to satisfy the short times required by the production. In such a situation the 60% of recipes were subject to delay with respect to the due time, and that was no more bearable. Moreover the queues before the machines were too long and presented a long waiting time.

Few months after, both an automatic machine to obtain colours and an automatic dyeing machine were acquired.

There the requests by the managers were the following:

- a) Evaluate the laboratory performance in the new structure compared to the old one;
- b) Suggest possible new structures and new managing solutions to increase such a performance, by evaluating costs and benefits of every new possible structure and solution. New structures might include new machines, both of the same type of the present one and of other types, and might eliminate some of the present machines. New managing solutions might change operating methods, turns, number of operators, scheduling rules.

MODELLING AND SIMULATION TECHNIQUES

As may be easily seen from dyeing laboratory description, the system to be studied is rather complex, it is subject to random requests and its behaviour depends upon scheduling rules which may be changed by the manager. Therefore discrete simulation was the instrument chosen to analyse the results obtainable by various solutions suggested for laboratory structure and management in order to determine the most convenient one. The chosen language was Micro Saint, which offers a good compromise among all simulation package characteristics: in fact it results comparatively

simple both to the model builder and to the model final user because of CAD and animation, but in the meantime it presents a good level of generality and abstraction. More precisely the language uses a small number of block types, respectively representing tasks, decisions and queues, connected by arcs, representing possible entity movements; each block may be equipped with functions ruling operating conditions and data collecting; both blocks and entities which move among blocks and stop inside them are represented by simple icons. This essential model presentation and the consequent simple animation implied some initial effort required to the model builder, but revealed to be a clear and quick communication mean. Moreover it permitted to build models characterised by a high level of modularity, which lead to a good standardisation level with consequent building time reduction.

LABORATORY GENERAL MODEL

A first phase was dedicated to study the laboratory past, current and future features and behaviours. A high level of detail was adopted, in that every single operation performed in the laboratory was accurately analysed and quantitatively described; every operations revealed a cycle time distributed according either to a normal or to a gamma function with proper parameters. Then a general flexible model was built up, including ten modules, each one describing a laboratory activity (including many single operations), as follows:

Module 1: sample generator (three Micro-Saint blocks)

Module 2: receipt design (ten blocks)

Module 3: manual colour preparation (nine blocks)

Module 4: colour preparation by automatic machine (thirteen blocks)

Module 5: colour solution set-up by robotised machine 1 (thirteen blocks)

Module 6: colour solution set-up by robotised machine 2 (seventeen blocks)

Module 7: manual yarn dyeing by simple machine (twenty two blocks)

Module 8: skein yarn dyeing by robotised machine (thirteen blocks)

Module 9: distaff yarn dyeing by robotised machine (ten blocks)

Module 10: sample check (eight blocks).

Obviously each module may be dynamically calibrated according to any specific operation performed (e.g., the mean operation time in module 6, 7 and 8 depends on the yarn type). Moreover possible decision may be ruled either by probability (e.g., quality check results) or by deterministic tactical choice (e.g., priority scheduling).

Module definition and simulation permitted to obtain a good accuracy in the system description and moreover to get good flexibility; in fact we could effect modifications both in the whole model, by including or cancelling some modules, and in any activity description, by the adjustment of one module only, without any change neither in other modules nor in the interfaces.

LABORATORY SPECIFIC MODELS

By properly combining modules above, it was possible to build up five different laboratory models, corresponding to five different past, current and future laboratory structures and related managing solutions, as follows:

- Model 1, including modules 1, 2, 3, 7, 10, describing the original, old structure where all operations were manual; a manually ruled machine was employed for 16 hour a day to prepare the colour and five manually ruled machines for 16 hours a day to dye samples; operators were 7 and the mean number of checks per sample were 4;
- Model 2, including modules 1, 2, 4, 7, 8, 10, describing the first enhancement of the old structure where some dyeing operations were manual and some were performed by a robotised machine with ten pools; with respect to the previous model a new automatic machine to prepare solution was employed for 24 hours a day and a new robotised machine with ten pools was added and use for 24 hours a day to dye samples; operators were 7 and the mean number of check per sample was reduced to 3, because of better behaviour of dyeing robotised machine;
- Model 3, including modules 1, 2, 5, 8 (doubled), 10, describing a second suggested enhancement of the structure where dyeing operations are performed by two robotised machine with ten pools; the number of operators is reduced to 6.5;
- Model 4, including modules 1, 2, 6, 8 (doubled), 10, describing a third suggested enhancement of the structure where colour solution set up is made by a more powerful machine and dyeing operations are performed by two robotised machine with ten pools; the number of operators is reduced to 6.5;
- Model 5, including modules 1, 2, 5, 6, 8 (doubled), 9, 10, describing a fourth suggested enhancement of the structure where colour solution set up is made by two machines and dyeing operations are performed by two robotised machine with ten pools and a third robotised machine with eight pools, able to dye yarn in distaffs; the number of operators is reduced to 6.5.

When implementing all models the input is always the same: a mean of 800 samples per week of different types; the outputs are given by the mean cycle time, the queues, the resource utilisation coefficients for all machines. On the basis of results the related economical considerations about conveniency were developed.

Model 1 was built up and simulated only for comparing purposes.

Model 2 was validated by comparing actual and simulated behaviours: the actual external demand related to 19 working weeks was given as input to the simulated model, and then the simulated trend of performed dyeing tests was compared with the trend of actual ones, obtaining a very good agreement: more precisely the actual output of every week was inside the range (mean \pm 3 std dev) obtained by 30 simulation runs. That proved the good accuracy of built up models.

SIMULATION RESULTS

By moving from Model 1 to Model 5, cycle time reduced from 28 hours up to 16 hours, with a 40% reduction; note that a 30% reduction was already obtained by Model 2.

Queues before performing operations are strongly reduced both for length and time; an exact comparison cannot be done as used machines are different.

Utilisation of dyeing pools increased up to 40% for manual dyeing machines (Model 2) and up to 90% for automatic machines (Models 2 to 5).

The effect of better results on production was a 25% reduction of bad samples.

From the economical point of view a total cost reduction of 45% was revealed by moving from Model 1 to Model 5, due to savings in costs related to employed operators and number of performed tests.

As a conclusion, modelling and simulation revealed to be extremely useful to precisely evaluate structural and managerial changes and related advantages for what concerns costs and overall laboratory behaviour.

REFERENCES

Drury E. C. and Laughery K. R., 2000, "Fundamentals of Simulation Using Micro Saint", Micro Analysis & Design Simulation Inc., Boulder, Colorado.

MICRO-SAINT 3.1: User's Manual, 2000, Micro Analysis & Design Simulation Inc., Boulder, Colorado.

BIOGRAPHY

GIORGIO ROMANIN-JACUR was born in Padova, Italy and went to the School of Engineering of the University of Padova, where he obtained the Laurea degree (equivalent to Master of Science) in Electrical Engineering in 1970 with a thesis about pulse modulation control system. He was fellow of the Italian Council of Researches in 1970-71; after military service in 1971-72 he was Research Associate of the same Council until 1980; in 1980-83 he was lecturer of Operations Research at the School of Engineering of the University of Padova; in 1983-2000 he was Associate Professor and since 2000 he is Full Professor of Operations Research at the same School. He currently teaches to the class of Management Engineering and to some Master and Doctorate classes in the Schools of Engineering, Medicine and Agriculture. His current researches concern production scheduling in the fields of Industry and Services.

Materials

Geometrical Modelling of Woven and Knitted Fabrics for Technical Applications

Xiaogang Chen and Xuegong Ai
Department of Textiles
UMIST, PO Box 88, Sackville Street
Manchester, UK M60 1QD
Email: Xiaogang.Chen@umist.ac.uk

KEYWORDS

Geometry, modelling, woven, knitted

ABSTRACT

The work on geometrical modelling of woven and knitted fabrics presented in the paper is part of a Technical Textiles CAD project, which aims to predict the mechanical and other properties from fibre properties and fabric structure for technical applications. In this work, geometrical models of single-layer woven fabrics and plain, rib, purl and interlock weft knitted fabrics are created based on Peirce's model of plain weave and its derivatives, Peirce's plain stitch model, Leaf's and Glaskin's plain stitch model and Leaf's elastic plain stitch model. Based on these geometrical models of woven and knitted fabric, WeaveGeoModeller and KnitGeoModeller have been developed to design fabrics, to calculate fabric geometry and some properties, to simulate load-on and filtration behaviours of woven fabrics, and to visualize 3D images of fabrics.

INTRODUCTION

The CAD technology has been used in the textile industry for many years, in systems covering the design of woven fabric, knitted fabric, braided fabric, and fashion and clothing. They are quite successful in style and pattern designing and help to ensure performance, comfort, fitting and endurance of the products. Nevertheless, the demand for higher performance of textile products is endless. The textile industry hopes to find a way to predict and optimise properties of fabric, in order to gain higher performance fabric for clothing and other domestic usage. Nowadays, fabrics are used not only for domestic usage, but also for industry use, for instance as composite reinforcement; it becomes a technical textile in the latter. Precise prediction of the properties of technical textiles is very important in these applications. To meet such needs, it is necessary to create a CAD system to predict the mechanical and other properties from fibre properties and fabric structural parameters. Physically, all of them are integrated properties, behaviours of and interactions between yarns in the fabric, and can be calculated if the fabric geometry, which is all shapes of yarns in the fabric, yarn properties and yarn mechanical state of that geometry are known. Fabric geometry is determined by the structure and the yarn properties, and the initial mechanical state of yarns in the on loom. Theoretically, it is possible to

calculate the geometrical shape of fabric with mathematical and mechanical models, but it can be very difficult, because of the structural complexity of yarns and fabric. To avoid these difficulties, some authors, such as Peirce [1,2], A. Kemp [3], B. Hamilton [4], John W. S. Hearle *et al* [5], M. Y. Sharkas [6], Dalidovitch [7], Leaf [8,9], Glaskin [8], and Smirfitt [10] had proposed geometrical models of woven and knitted fabrics, respectively, by using several types of curve, such as line, arc or other curve or their combinations to represent the yarn path in the fabric approximately. With the development of computer graphics, researchers began to try to represent the fabric geometry on the screen for CAD systems. H. Y. Lin [11], X. Chen and P. Potiyaraj [12, 13, 14] work on the representation of woven fabric. Michael Meissner *et al* [15], A. Demiroz and T. Dias [16] generated 3D graphical representations of selected knitted fabrics. Most of their works correctly represented fabric topological structures but lack of concern of fabric properties. In this paper, we introduce the geometrical models for woven and knitted fabrics based on some previous work and the visualization packages WeaveGeoModeller and KnitGeoModeller, developed through the Technical Textiles CAD projects.

GEOMETRICAL MODELLING OF WOVEN FABRICS

Parameters used to construct models

The objective here is to construct a geometrical woven fabric model with a number of controllable fabric parameters sufficient to determine the geometry of woven fabrics. The weaves (fabric structure), structures of parameters are select as parameters to specify a woven fabric. Among these parameters, the fabric structure and yarn type and yarn count are selected at the fabric design stage, and the fabric densities and crimps can be controlled by appropriate loom setting and they are also the parameters which can be used to solve functions in Peirce's model and its derivatives.

Peirce's model and its derivatives

The basic structural element of woven fabric is the interlacement. In a plain-woven fabric, the cross-angle of a warp yarn and a weft yarn is 90°. Using this premise and another assumption that yarns have a uniform cross-section, Peirce [1] assumed that at an interlacement, one yarn path follows the cross-section shape of the other yarn, and between two neighbouring interlacements, the yarn takes the shortest path: the straight line segment. He assumed the

yarn to be a column with a uniform cross-sectional shape along its length whether it is straight or bent. According to the assumptions of yarn and the path, the diagram can be drawn as shown in Figure 1.

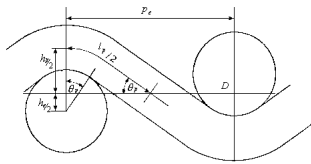


Figure 1 Peirce's model

Figure 1 is a cross-section of one repeat (cell) of plain-woven fabric through the middle of a weft yarn. Because yarns flatten where they interlace, other cross-sectional shapes were considered. A. Kemp's racetrack model [3] and John W. S. Hearle and W. J. Shanahan's lenticular model [5] are the most widely used two models. Derived from Peirce's model, their yarn paths follow the same assumption as those of his circular model.

Woven fabric model

Peirce's woven fabric model and its derivatives are reasonably accurate for plain-woven fabrics. However, the interlacements in the non-plain woven fabrics are some different. There are horizontal (along the warp or weft direction) bending and vertical (along the fabric thickness direction) bending of a yarn in the non-plain woven fabrics as pointed out by Hamilton [3] and K. Balakrishna Iyer, etc [17], though in many scenarios the horizontal bending is not considerable. By comparing the yarn path of non-plain woven fabrics and the yarn path of Peirce's model, it can be seen that the bending of yarn is the same or symmetrical when the yarn crosses the neutral plane of the fabric in the intersection spacing either in a plain or a non-plain woven fabrics. This has been illustrated in Figure 2. In this context, we assume that the yarn path in the intersection spacing of non-plain woven fabrics satisfies Peirce's model and its derivative models, and that the yarn path in the float spacing is a straight-line segment and the intersection spacing and the float spacing are the same or symmetrical. Then, combining the fabric structure (the interlacement arrangement of a fabric) all the yarn paths within a repeat can be calculated and the fabric geometry can be obtained.

WeaveGeoModeller

Based on the geometrical modelling of woven fabrics, under windows platform, we develop a visualization package for designing and simulation single layer woven fabrics, using Delphi 5.0 and OpenGL. Figure 3 shows the architecture of WeaveGeoModeller. Using the geometrical models of woven fabrics, WeaveGeoModeller can calculate the fabric geometry and visualize the fabric on the screen, then calculate fabric properties such as the porosity, thickness, and weight per unit area for woven fabrics. A load-on deformation model was created to animate the deformation of a fabric under tensile load and a filtration model is created to simulate the situation when particles are filtrated by the fabric. The mechanical model [18] used to calculate the deformation of the fabric is loaded as a dynamical link

library develop by another research student cooperated in the same project.

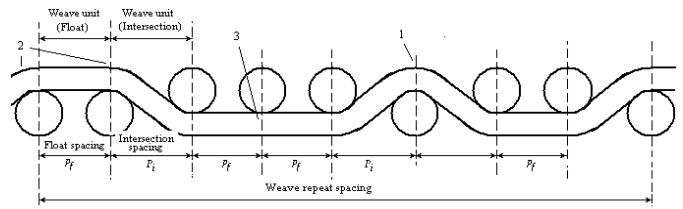


Figure 2 Weft cross-section of 2/3/1/2 twill

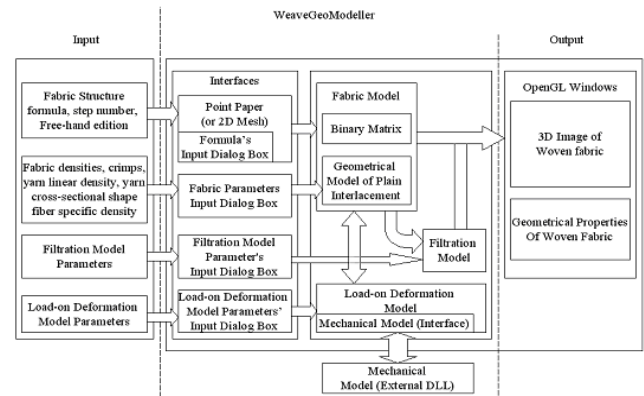


Figure 3 Architecture of WeaveGeoModeller

Specification of woven fabric in WeaveGeoModeller

Woven fabrics are specified by the weave (fabric structure), properties of yarns used and fabric parameters. In WeaveGeoModeller, we use point paper, 2D mesh and formula methods to input it. The point paper in WeaveGeoModeller is an editing tool to create and modify woven structures, which are inherited from Pranut Potiyaraj's work [19]. The 2D Mesh method was developed and implemented in the program as weave creation tool for those who may not be familiar with use of the point paper. The difference between the 2D mesh and the point paper is that the 2D mesh presents the fabric in terms of yarn mesh. Figure 4 (a) and (b) shows an example of point paper and 2D mesh presentation of a weave respectively.

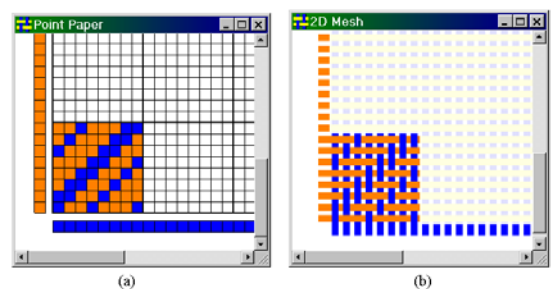


Figure 4 A weave displayed on (a) point paper and (b) 2D mesh

The formula method is a relatively new method created by X. Chen *et al* to input woven fabric structures [20]. Figure 5 shows the dialogue box implemented to input such a formula. The dialogue box in Figure 5 displays the

formula $\frac{2}{3} \frac{1}{2}$ at the upper part of the box, which is specified in the form of 2/3/1/2.

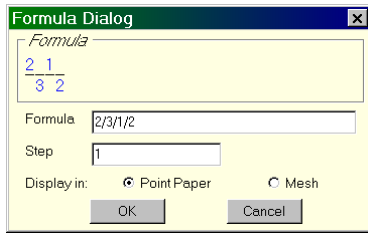


Figure 5 Dialogue box to input a formula

According to the Peirce's model and its derivatives, arbitrarily specified fabric densities values and yarn crimp values will not necessarily lead to reasonable fabric geometries because of the geometrical restriction of fabrics. Therefore, the program provides practical ranges for these two groups of parameters as guidance to specifications. When a user inputs or modifies the fabric densities and yarn crimps, the guidance information will be displayed underneath the text field shown in Figure 6.

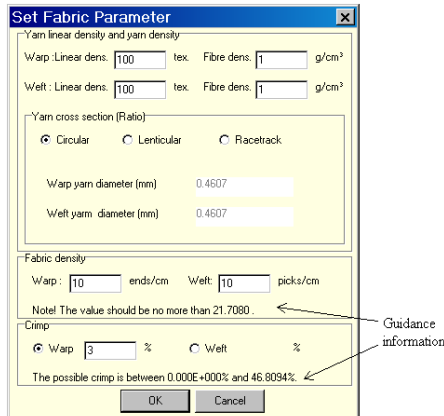


Figure 6 Dialogue box to set up fabric parameters

Examples

Figure 7 shows examples of 3D images of loose plain-woven fabric of different cross-sectional models. Figure 8 shows 3D images of non-plain-woven fabric.

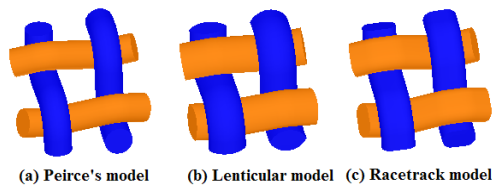


Figure 7 Plain-woven fabric cells

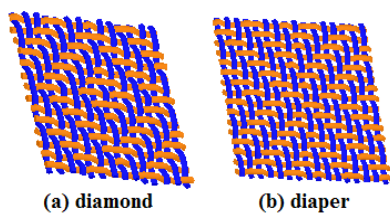


Figure 8 Non-plain woven fabrics

GEOMETRICAL MODELLING OF KNITTED FABRICS

Geometrical model of knitted fabrics

The basic structural element of a knitted fabric, known as the stitch, is of several different types, such as plain, float, tuck, twist stitch, and so on. The structural element of the plain, rib, purl and interlock knitted fabric is the plain stitch, which can be R-loop or L-loop. By arranging R-loop and L-loop plain stitches in different ways, different types of knitted fabric can be obtained. On the face of fabrics, all stitches are R-loop in plain knitted, while R-loop and L-loop stitches appear as a whole wale cord in turns along the course direction in a rib structure and R and |L loops run alternatively as a whole course along the wale direction in purl. Interlock knitted fabric is comprised of two layers of ribs interlocked together with crossed sinker wales. The 3D image of plain, rib, purl and interlock knitted fabrics is obtained by drawing the R-loop or L-loop plain stitch in the corresponding place according to the fabric structure.

Geometrical models of a stitch

Besides the fabric structure, we need to know the geometry of plain stitch to determine knitted fabric geometry. Three geometrical models of plain stitch were used in this paper. They are Peirce's plain stitch model, Leaf and Glaskin's model, and Leaf's Elastic model. In these models, arcs and straight line segments, arcs and elastica are used to represent the orthogonal projection of stitch paths on the fabric plane respectively. The side projection of stitch path is assumed to be an arc, sine curve or cosine curve. Figure 9 shows 3D images calculated from these three models.



(a) Peirce's model (b) Leaf and Glaskin's model (c) Leaf's elastic model

Figure 9 Stitches of Different Models

Modifications were necessarily made to parameterise these models. The parameters used are the wale spacing, course spacing, and the yarn diameter.

Interlock stitch model is created based on these three models by modifying slightly yarn path in order to make the yarn path of two neighbouring R-loop or L-loop stitch connected smoothly.

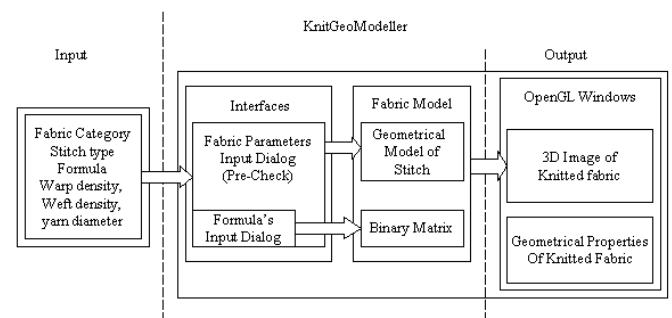


Figure 10 Architecture of KnitGeoModeller

KnitGeoModeller

Based on the geometrical modelling of knitted fabrics, KnitGeoModeller has been designed to implement three main functions: (i) to allow users to input parameters of knitted fabric, fabric structure type, yarn diameter, course density, wale density, (ii) to calculate the geometry and other geometrical properties of the fabric, *i.e.* the porosity, thickness and stitch length for knitted fabrics and (iii) to visualize the fabric. The architecture of KnitGeoModeller is shown as in Figure 10.

When the fabric structure, model type, side projection type, warp and course densities and yarn diameter are specified, KnitGeoModeller calculates the geometry and properties of the fabric. In KnitGeoModeller, there are four types of fabric structure, plain, rib, purl and interlock can be simulated. Similarly to woven fabrics, the formula is used to describe rib and purl structure. Figure 11 displays the interface used for fabric specification.

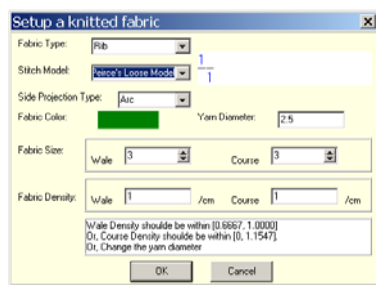


Figure 11 Dialogue box to set up knitted fabric parameters

Examples

Figures 12 (a), (b), (c) and (d) show 3D images of plain, 1/1 rib, 2/2 purl, and interlock knitted fabric using the three different stitch models.

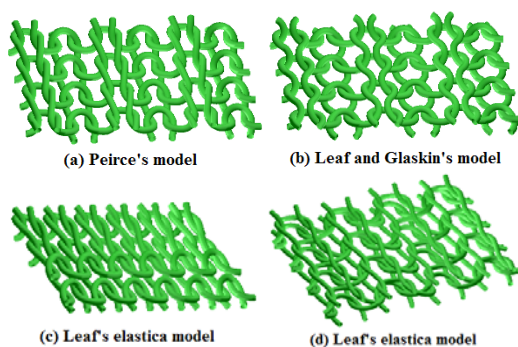


Figure 12 3D images of knitted fabric

CONCLUSIONS

Using Peirce's models and its derivatives for woven and knitted fabrics, we have constructed the geometrical models of woven and knitted fabrics based on mathematical modelling, and hence WeaveGeoModeller and KnitGeoModeller are developed. Upon the specification of fabric structure and fabric parameters, the packages can be used to create woven and knitted fabrics, calculate their geometry, visualize fabrics, predict some properties of

fabrics, and simulate some behaviours, such as filtration behaviour and fabric deformation when load is specified.

REFERENCES

1. Peirce F. T., The Geometry of Cloth Structure, J. Text. Inst. Transaction, T45, 1937
2. Peirce F. T., Geometrical Principles Applicable to The Design of Functional Fabrics, Text. Res. J., 17, p123, 1947.
3. A. Kemp, an Extension of Peirce Cloth Geometry to the Treatment of Noncircular Threads, J. Textile Inst., T45, 1958.
4. J. B. Hamilton, General System of Woven-Fabric Geometry, J. Text. Inst., T66, 1964.
5. W. J. Shanahan and J. W. S. Hearle, An Energy Method for Calculations in Fabric Mechanics Part II: Examples of Application of The Method to Woven Fabrics, J. Textile Inst., 69, No.4, p92, 1978.
6. M. Y. Sharkas, PHD Thesis, University of Leeds, 1981.
7. Dalidovitch, A. S., Osnovy Teorii Wjazaniya, Gizlegprom, Moscow, 1949.
8. G. A. Leaf, A Glaskin, The Geometry of Plain Knitted Loop, J. Textile Inst., p587, 1955.
9. G. A. Leaf, Models of the Plain-Knitted Loop, J. Textile Inst., T49, 1960.
10. J. A. Smirfitt, Worsted 1x1 Rib Fabrics Part I. Dimensional Properties, J. Textile Inst., 1965.
11. H. Y. Lin and A. Newton, Computer Representation of Woven Fabric by Using B-splines, J. Textile Inst., Part 1, No. 1, 1999.
12. X. Chen, P. Potiyaraj, CAD/CAM for Complex Woven Fabrics Part I: Backed Cloths, J. Textile Inst., Volume 89, p532, 1997.
13. X. Chen, P. Potiyaraj, CAD/CAM for Complex Woven Fabrics Part II: Multi-layer Fabrics, J. Textile Inst., Volume 90, p73, 1997.
14. X. Chen, P. Potiyaraj, CAD/CAM for Orthogonal and Angle-Interlock Woven Structures for Industrial Applications, Text. Res. J., 69(9), 648, 1999.
15. Michael Meissner, Bernhard Eberhardt and Wolfgang Strasser, A Volumetric Appearance Model, 1996.
16. A. Demiroz and T. Dias, Part I: Stitch Model for the Graphical Representation of Plain-knitted Structures, J. Textile Inst., p463, part 1, No. 4, 2000.
17. K. Balakrishna Iyer, K. Hepworth, and D. C. Snowden, The Inherent Skewness of 2/2-Twill Weave, J. Textile Inst., T99, 1964.
18. Vidya Sagar Thammandra, Mechanics of Woven preforms and textile composites, first year report, 2002.
19. Pranot Potiyaraj, Mathematical Modelling and Development of a CAD/CAM System for 3D Woven Structures, Ph.D. thesis, UMIST, 2000.
20. X. Chen, R.T. Knox, D.F. McKenna, and R.R. Mather, Automatic Generation of Weaves for the CAM of 2D and 3D Woven Textile Structures, J. Textile Inst., Volume 87(2), Part 1, 356, 1996.

FRACTURE TOUGHNESS OF FLOCK FIBER REINFORCED LAYERED COMPOSITES

Yong K. Kim
Department of Textile Sciences
University of Massachusetts Dartmouth
Dartmouth, MA 02747 USA
ykim@umassd.edu

Sunil Hoskote
Guilford Mills, Inc.
141 Wideawake Street
Pine Grove, PA 17963
USA

KEYWORDS

Micromechanical Fracture Model, Mode I Fracture, Organic Laminar Composites, Double Cantilever Beam

ABSTRACT

The effect of reinforcing flock fibers, as through-thickness z-directional fibers, on the Mode I fracture toughness property of layered glass fabric composites in an epoxy resin matrix is reported. Three types of flock fibers, 20 denier nylon, 3 denier nylon and milled glass flock are studied. A simple micro-mechanical model is proposed to study the fracture behavior of flock fibers in a double-cantilever-beam specimen for mode I fracture. The mode I interlaminar fracture toughness of z- reinforced flocked composites shows an improvement of up to 10 times the non-flocked standard layered polymer glass fabric composites.

Overall, z-directionally reinforced laminates with thermoplastic nylon flock fibers shows the most significant improvement in delamination fracture toughness. Finally, the developed micro-mechanical fracture model was found to fit the experimental data reasonably well.

INTRODUCTION

Delamination of layered fabric-reinforced organic polymer composite materials represents one of the most prevalent, structural, life-limiting failure modes. The growth of these delaminations results in a loss of stiffness and could eventually result in catastrophic failure. Hence it is of great importance that the interlaminar fracture toughness of laminated fabric/resin composites be high and have good fatigue durability. Improved resistance to delamination should guarantee extended service life and improved mechanical performance of these advanced materials.

Various techniques have been introduced to enhance the interlaminar strength of layered composite materials (Gillespie 1991, Morley 1987). A common technique is to use rubber-toughened matrix material resin (Takeda et. Al. 1999). However these resins are not thermally durable. An alternative approach, which is gaining popularity, is to manufacture special fabrics using advanced textile technologies such as knitting, 3-D weaving or a through-the-fabric stitching process (Tanzawa et al 1999). These methods are slow and result in 3-D orientation in the fabric. These orientations are usually not conducive to optimized stress distribution in fabricated mechanically functioning structural components. Such 3-D structures are prone to stress concentrations under mechanical service leading to poor fatigue resistance. All these approaches work in their

primary goal, but they degrade the composite's in-plane properties (Cox et. al. 1997).

In this paper, studies are reported on developing a method of fabricating a composite with improved delamination resistance by interlaminar z-directional flock fiber placement. The effect of this flock reinforcement on the mode I fracture toughness is examined with the help of a micromechanical model.

MICROMECHANICAL MODEL

A simple micro mechanical model is proposed to study the effect of z flock fiber through-thickness reinforcement on the delamination resistance in a double-cantilever-beam (DCB) specimen.

Model Assumptions Made

1. The flock fibers are cylindrical
2. The matrix-flock fiber interface only supports shear stress and the elastic bond strength is neglected.
3. The frictional shear stress at the matrix-flock fiber interface is assumed to have a constant value.
4. The deformation in the matrix is assumed to be negligible.

Geometry of the Double-Cantilever-Beam Specimen

The DCB model of through-thickness reinforced laminar composite is shown in Figure 1.

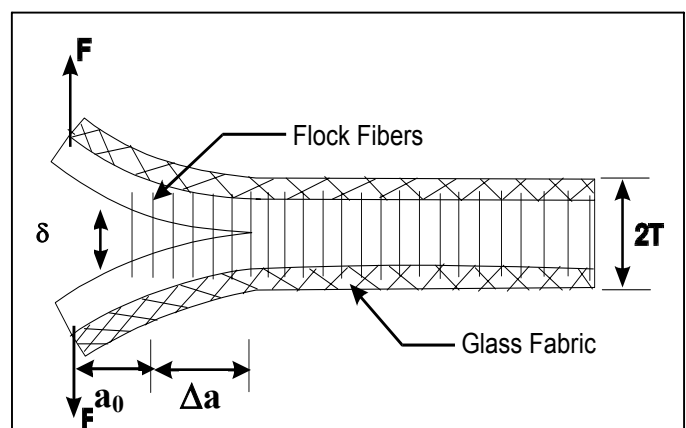


Figure 1: DCB Specimen Geometry Showing the Bridging Flock Fibers

The symbols used in the model are defined as follows:
 a_0 = initial delamination

Δ_a = crack growth characterized by flock bridging zone length
 T = half beam thickness
 F = applied Load per unit width of the beam
 $\delta(t)$ = the displacement at the point of the load
 d_f = flock fiber diameter.

It was determined by microscopic measurement that on an average all the flock fibers penetrate 1/10th of the fabric thickness. There is n number of fibers per unit width of the DCB specimen. The flock density, n , is obtained experimentally. If the embedded length of the fiber (T) is smaller than the critical length $L_c (= d_f \sigma / 4\tau)$, the fiber will be initially stretched and subsequently pulled-out with increasing crack-opening displacement. Here τ is the shear stress at the matrix-flock fiber interface. However, if $T \geq L_c$ the flock fibers will just stretch and then rupture when the stress in the fiber at the crack surface equals the tensile strength of the fiber.

Flock Fiber Motion During Crack Propagation

Let us consider a single flock fiber embedded in a matrix with embedded length T as shown in Figure 2. “ Y is the portion of the fiber carrying the load before pull-out of the fiber begins. The slippage distance S is the slippage of the embedded end after pull-out of the fiber begins.

As the load “ F ” increases, the flock fiber begins stretching and the slip length increases. This extension of the slip length is the debonding process. This slip length increases from zero to “ T ” before the embedded end starts slipping out. During this process, the load carried by the flock fiber increases from zero to its maximum.

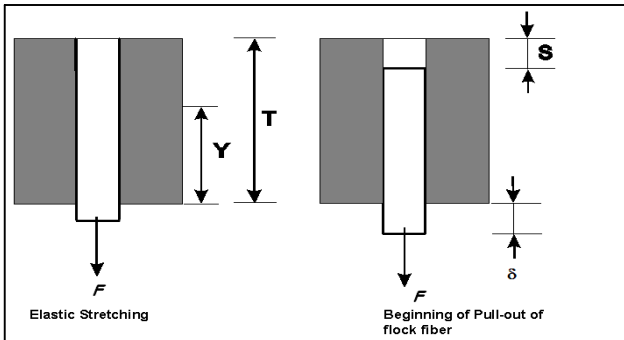


Figure 2: Pull-Out of Embedded Flock Fiber

The displacement “ δ ” of the loaded end of DCB also increases as a result of the stretching of the fiber segment. Provided that the slip length reaches the value “ T ” without fiber rupture, the fiber pullout proceeds and the load carried by the fiber decreases.

A force F versus displacement δ relation was derived by Jain and Wetherhold (Jain and Mai 1994, Jain and Wetherhold):

During elastic stretching,

$$F(Y) = \tau \pi d_f Y \quad (1)$$

$$\delta(Y) = \left[Y - \frac{T}{r} \ln \left(\frac{Yr}{T} + 1 \right) \right] \left[1 + \frac{Yr}{T} \right] \quad (2)$$

During pull-out,

$$F(T) = (T - S) \tau \pi d_f \quad (3)$$

$$\delta(T) = \left[T - \frac{T}{r} \ln \left(\frac{(T - S)r}{T} + 1 \right) \right] \left[1 + \frac{(T - S)r}{T} \right] \quad (4)$$

Both slip length “ Y ” and slippage “ S ” increase from “0” to “ T ”. The extensibility ratio “ r ” (Jain and Wetherhold 1992), which characterizes the composite is defined as:

$$r = \tau \frac{\pi d_f T}{A_f E_f} \quad (5)$$

Where ‘ A_f ’ is the cross-sectional area and ‘ E_f ’ is the Young's modulus of the flock fiber. Equations (1) and (2) give the rising portion of the F - δ curve and Equations (3) and (4) give the declining portion of the F - δ curve.

Composite Fracture Energy

The contribution of the bridging flock fibers to the fracture energy can be determined by finding the energy contribution due to elastic stretching and pull-out of a single fiber and then integrating the contributions from all the fibers. (Jain and Wetherhold 1992, Reedy et al 1997)

$$g(T) = \int_{Y=0}^T F(Y) \frac{d\delta(Y)}{dY} dY + \int_{S(0)=0}^T F(T) \frac{d\delta(T)}{dS(0)} dS(0) \quad (6)$$

Making substitutions from equation (1), (2), (3) and (4) and integrating we obtain the fracture energy for n fibers per square millimeter.

$$g(T) = Ge = n \tau \pi d_f \frac{T^2}{2} \left[1 + \frac{r}{3} \right] \quad (7)$$

The cohesive fracture energy for unmodified epoxy resin,

$$Gc = \pi a \sigma^2 / E \quad (8)$$

Where, a is crack length, σ is the tensile strength and E is the tensile modulus of the resin, respectively.

The strain energy released due to the bending of a cantilever beam is given by:

$$U = Gb = \frac{1}{2EI} \int M^2(x) dx \quad (9)$$

where, E is the modulus of the beam, I is the moment of inertia of the beam section, and $M(x)$ is the bending moment at crack length x . The integral limits are a to $a + \Delta a$.

The total fracture energy, $G(T)$, released from the DCB specimen is given by:

$G(T)$ = fracture energy due to bridging flockfibers (Ge) + cohesive fracture energy for epoxy resin (Gc) + energy released due to bending of the beam (Gb).

Substituting equations (7), (8) and (9), we get the strain energy released by the flocked composite:

$$G(T) = n \tau \pi d_f \frac{T^2}{2} \left[1 + \frac{r}{3} \right] + \pi a \sigma^2 / E + \frac{1}{2EI} \int M^2(x) dx \quad (10)$$

Therefore, the fracture toughness of flocked composites is determined to be the sum of: (1) the fracture energy of the bridging flock fibers (elastic stretching and pull-out), (2) cohesive fracture energy of the bulk epoxy resin and (3) energy released due to the bending of the beam. The fracture energy contribution of the bridging flock fibers is calculated by integrating from $Y = 0$ to T of the elastic stretching of the fiber in the matrix and adding the pull-out

contribution of the fibers by integrating the length of the flock fiber from $S = 0$ to T . From Figure 2, Y equals the length of the fiber in the matrix at the time of elastic stretching, T , the a length of the initial flock fiber and S , the slippage length of the embedded end of fiber.

EXPERIMENTAL VERIFICATION

Materials

Woven glass fabric of $0.335 \text{ Kg} / \text{m}^2$ was obtained from *Fibre Glast Inc.* Brookville, Ohio. The matrix (adhesive) material used is the amine cured epoxy resin (*Shell Epicure 3223* curing agent and *Shell Epon 826* epoxy resin.) The flock fibers considered for the study are: 20 denier, 2.54 mm long, nylon flock, 3 denier, 0.25 mm long nylon flock, 0.79 mm long milled glass fiber flock. The nylon flock fibers were supplied by *Claremont Flock Corporation, Leominster, MA* and the milled glass from *Fibre Glast*.

Fabrication of Z-Reinforced Organic Laminar Composite

Amine cured epoxy resin was used as the flocking adhesive. The glass fabric was wetted by the matrix/adhesive resin to make sure the resin penetrated the interstices of the glass fabric. After the application of the matrix/adhesive resin, the fabric is flocked by the electrostatic flocking method. The flocking device used was a laboratory hand flocking machine *Dekor®* produced by Creative Coatings Corporation. In this method, the uncured matrix/adhesive resin coated substrate passes between the potentials of a high voltage electrostatic field. An electrode gives the flock a charge and the charged fibers become aligned with the electric force field. The substrate and/or the grounded parts of the base plate form the ground potential. The flock is thus attracted to the still fluid adhesive layer where it gets embedded. After flocking the glass fabric, it is allowed to stand for 5 minutes while the epoxy resin starts curing. The next step is to draw the flock fibers into the interstices of the glass fabric by vacuum bagging the flocked glass fabric. Vacuum bagging applies an amount of pressure that allows the fibers to embed themselves into the substrate. The vacuum bagging process is carried out for 15 to 30 minutes. After this time, the flocked fabric is allowed to cure at room temperature for 12 to 14 hours.

Three types of laying sequence are followed. In the first sequence called F1 (Figure 3 (a)) only one side of the glass fabric is flocked. In the second called F2 (Figure 3 (b)) both sides of the glass substrate is flocked. The objective of the F2 lay-up experiment is to increase the flock density and thus the volume fraction of the z-fibers in the laminated composite. The last type of laminated composite F3 is the one without any flock between the laminas. This type is used as control for comparison. The lay up sequence is shown in Figure 3 (c).

The F1/Epoxy, F2/ Epoxy, F3/Epoxy laminated composites were prepared using the wet lay-up procedure. The impregnation of the layers is done in process, that is, at the time the material is laid into the mold. Each layer is completely wetted with amine curing epoxy resin. The properties, the resin content of a composite laminate now

depends on the curing procedure. In the present experiment, the samples were cured at a temperature of 100°C in a flat press for one hour.

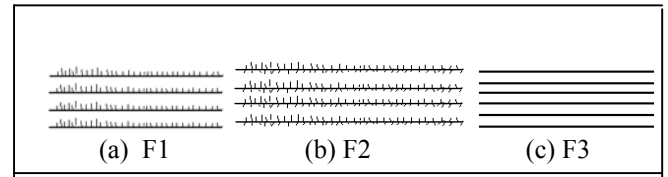


Figure 3: Laminates Stacking Sequence; (a) Flock Fibers on One Side of the Glass Fabric Substrate (F1), (b) Flock Fibers on Both Sides of the Glass Fabric Substrate (F2), (c) Standard Lay up (F3)for Comparison

In order to achieve approximately the same resin content in each type of laminas, the pressure applied is varied based on the thickness of the non-flocked and flocked glass fabric. First the thickness of the non-flocked fabric is measured and this is fabricated to a composite applying a pressure of around 555.5 kPa. Using this as a standard, the appropriate pressure required for F2 and F3 is calculated and is found to be 695.3 kPa for F2 and 766.4 kPa for F3, respectively.

Mode I Fracture Toughness

TestDouble Cantilever Beam (DCB) test, ASTM D 5528 is used to determine the Mode I fracture toughness property. The test set up is schematically shown in Figure 4. The specimen is 127 mm long, 20 to 20.54mm wide and 3 to 5mm thick.

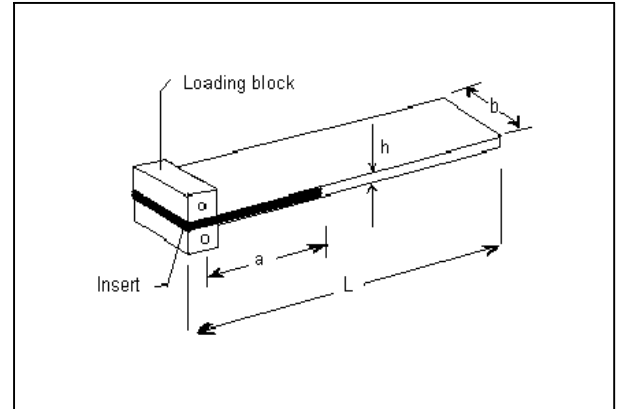


Figure 4: Double Cantilever Beam Specimen

A CCD high-speed camera system (*Cooke SensiCam* high speed camera with *Infocus KC* microscope lens with a magnification of 64x) is positioned on one side the specimen to observe and record the delamination front as it extends along one edge during the test. Load to the specimen is applied at a crosshead displacement rate of 0.5mm/min and the load versus displacement trace on an X-Y chart recorder.

G_{IC} at various delamination length is generated to characterize the initiation and propagation of a delamination (Lewis and Saxon 1973). G_{IC} is calculated from the load and displacement at the point of deviation from linearity as described in ASTM D5528.

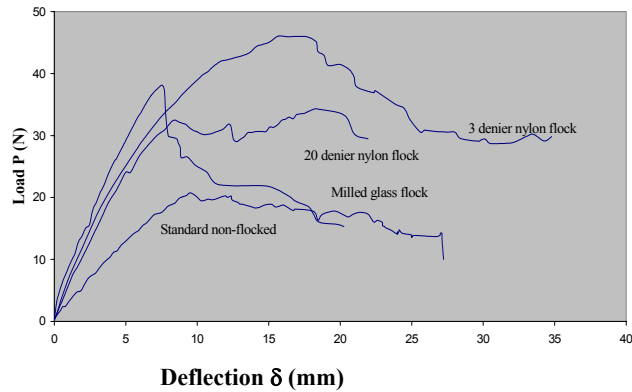
The strain energy release rate of a perfectly built-in double cantilever beam is calculated from Equation (11).

$$G_I = 3P\delta/2ba \tag{11}$$

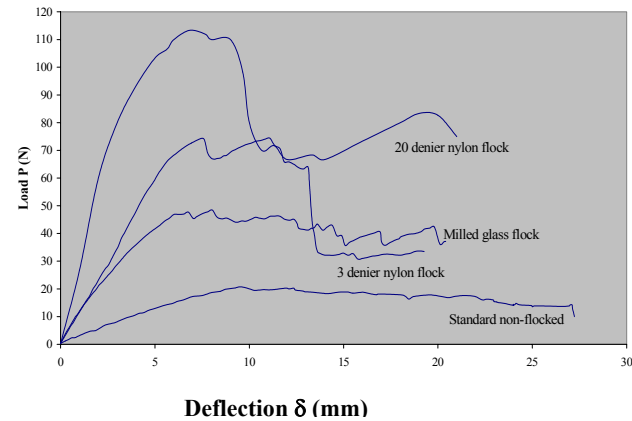
Where P = load (N), δ = load point displacement (mm), b = specimen width (mm) and a = delamination length (mm).

Interlaminar Mode I Fracture Toughness

Observations of the load-displacement curves, Figures 5 (a) and (b), obtained from flock reinforced composites show that initially there is an increase in the crack propagation load as the crack grows. This is caused by the development of the flock-bridging zone. After the bridging zone is fully developed, steady-state interlaminar fracture toughness is attained and then the load decreases with increasing crack length. The crack propagation part of the curve generally exhibits stable crack propagation behavior consisting of successive load drops followed by crack arrest. This is termed the "stick-slip" behavior. In the case of the standard non-flocked composite, there is a continual decrease in the crack propagation load with increasing crack length.



(a) Flocked on One Side of the Glass Fabric Substrate

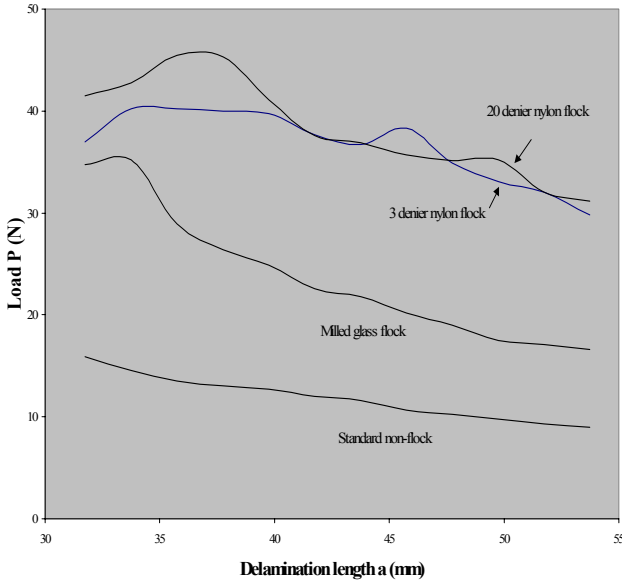


(b) Flocked on Both Sides of the Glass Fabric Substrate

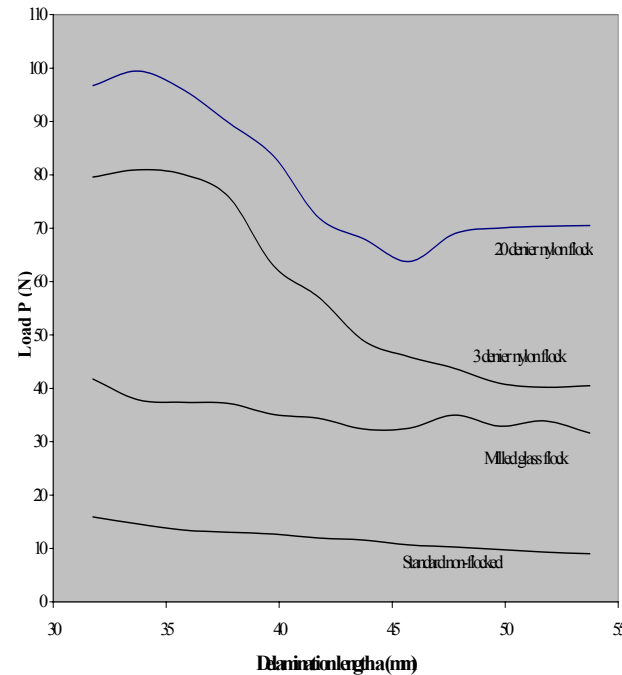
Figure 5: Load-Deflection Curves for Various Composites Obtained from Mode I Fracture Toughness Testing Using a DCB Specimen

Figures 6 (a) and (b) show load as a function of delamination length 'a'.

These curves show how the crack propagates, with the flock modified composites showing a stick/slip behavior.



(a) Flocked on One Side of the Glass Fabric Substrate



(b) Flocked on Both Sides of the Glass Fabric Substrate

Figure 6: Load-Delamination Length Curves for Various Composites Obtained from the Mode I Fracture Toughness Testing Using a DCB Specime

The average strain energy release rate G_I values for various composite lay-up sequence and flock fiber type are given in Table 1.

Table 1: Mode I Fracture Toughness of Various Types of Composites Based on ASTM Standard D5528

Lay-up	Flock type	Flock Denier (length)	Flock fiber density (fibers/mm ²)	Flock fiber volume fraction (%)	Total fiber volume fraction (V _f)	G _I * (KJ/m ²)
F1	Nylon	20 denier (2.54mm)	8	7	70	0.96
		3 denier (0.254mm)	346	4.7	70	1.54
	Milled glass	0.793 mm milled glass	N/A	12.1	69	0.83
F2	Nylon	20 denier (0.254mm)	27	12	70	2.24
		3 denier (2.54mm)	1244	8.5	68	2.15
	Milled glass	0.793 mm milled glass	N/A	15	68	0.99
F3 (No flock)					72	0.28

*G_I value taken at the point where the load-deflection curve becomes non-linear, where the ASTM formula $G_I = 3P\delta/2ba$ is used.

The individual values for each type of laminate are given in Tables 2 and 3. It is obvious from the data that the addition of through-thickness reinforcement in the form of flock fibers produced a significant improvement in the Mode I delamination fracture toughness.

The measured fracture toughness shows however that the G_I value does not remain constant, and this is because the delamination grows as a slip/stick process. In the DCB tests it is observed that, when the delamination approaches a row of flock fibers the crack growth speed is slowed until some of the fibers bridging the delamination zone fail and/or pull out under a higher load. This then allows the delamination to propagate rapidly to the next row of flock fibers where the process repeats itself. This slip/stick fracture process contributes to the variations in the G_I values of the flocked composites. When the elastic stretching of the flock fiber proceeds, the G_I value in the same series shows an increasing trend. Once the flock fiber is pulled out there is a drop in the resistance.

The flock reinforcement in the z-direction exerts a closure traction force, which resists to the crack opening stress acting on the crack tip. Some toughening can also occur by a deflection or blunting of the crack tip caused by the pullout of the flock fibers.

Table 2: Mode I Fracture Toughness for Various Single Side Z-directional Flock Reinforcement

Delamination Length (mm)	G _I (kJ/m ²)			
	Nylon 20 den	Nylon 3 den	Milled glass	Standard (Base)
31.75	0.96	1.54	0.83	0.28
33.75	1.48	2.28	1.19	0.27
35.75	1.60	2.69	1.03	0.29
37.75	1.91	2.73	0.97	0.29
39.75	2.07	2.61	0.97	0.31
41.75	2.21	2.63	0.89	0.32
43.75	2.30	2.71	0.87	0.34
45.75	2.63	2.70	0.82	0.34
47.75	2.40	2.81	0.81	0.33
49.75	2.46	2.95	0.77	0.32
51.75	2.44	2.86	0.76	0.32
53.75	2.37	2.92	0.77	0.32

Table 3: Mode I Fracture Toughness For Various Double Side Z-Directional Flock Reinforcement

Delamination Length (mm)	G _I (kJ/m ²)			
	Nylon 20 den	Nylon 3 den	Milled glass	Standard (Base)
31.75	2.24	2.15	0.99	0.28
33.75	3.09	2.74	0.94	0.27
35.75	3.11	2.94	1.01	0.29
37.75	2.98	2.98	1.10	0.29
39.75	2.73	2.50	1.10	0.31
41.75	2.50	2.31	1.09	0.32
43.75	2.53	2.03	1.05	0.34
45.75	2.60	1.93	1.09	0.34
47.75	3.12	1.94	1.21	0.33
49.75	3.40	1.87	1.24	0.32
51.75	3.55	1.90	1.33	0.32
53.75	3.58	1.94	1.26	0.32

The nylon flock reinforced composites show the characteristic behavior as described above. The glass flock fiber reinforced composites show rapid unstable crack propagation. This is because with glass fibers, there is little elastic stretching of the fibers. Also the milled glass fiber flock is quite short 0.793mm (1/32") in length, which is shorter than the critical length, L_c, for the fiber to be pulled out from the matrix resin.

The non-flocked, "standard" composites, having a constant surface energy of the more homogeneous, bulk epoxy, shows stable low crack propagation without any stick/slip behavior.

Theoretical Prediction of Mode I Fracture Toughness

The mathematical model, Equation 10, explains the behavior of flock fibers during a Mode I fracture. The flock fibers initially stretch resulting in a rising load-deflection curve and then the pullout of the flock fiber proceeds resulting in the drop in the load-deflection curve. This was demonstrated in Tables 2 and 3. The extensibility ratio, r , increases the bridging energy by the elastic stretching of the fibers. In turn this gives the increasing portion of the G_I values. Based on this model, it can also be seen that the higher flock density increases the Mode I fracture energy.

Table 4 compares the theoretical and experimental G_I values. These values were obtained for 20 denier nylon single side reinforcement at a delamination length of 45.75 mm, and for 3 denier nylon single sided at a delamination length of 49.75 mm, respectively. The model agrees reasonably well with the experimental data considering the experimental variability of the test results.

Table 4: Theoretical and Experimental G_I values

	Experimental G_I (KJ/m ²) (ASTM D5528)	Theoretical G_I (KJ/m ²) (Micro mechanical model)
20 denier nylon flock (Flock density 8 /mm²)	2.63	2.44
3 denier nylon flock (Flock density 346 /mm²)	2.95	3.49

CONCLUSIONS

The study on the effects of z-directional through-thickness flock fiber reinforced glass fabric/epoxy resin matrix composite on the delamination failure performance properties of organic polymer composites shows that:

1. The addition of through-thickness reinforcement in the form of flock fibers produces significant improvements in mode I fracture toughness. An improvement as high as 10 times in Mode I is observed as compared to the non-flocked, "standard", glass fabric/epoxy composites.
2. Increasing the flock density (by flocking on both sides) improves this Mode I fracture toughness. However, there is no significant difference (at significant level $\alpha = 0.05$) in mode I fracture toughness mean values between 20 denier and 3 denier nylon flocked composites.
3. Significant differences in the arithmetic mean of Mode I fracture toughness are observed between nylon flocked (both 20 and 3 denier) and milled glass flock. 20 denier nylon flock showed the highest improvement of up to 10 times. Milled glass flock showed the least improvement among the flock fibers tested of 4 times, as compared to the non-flocked, "standard" composite.
4. A load / deflection curve showing a characteristic stick-slip behavior is observed for the z-reinforced laminated composite. The flock reinforcement in the z-direction

exerts a closure traction force, which lowers the stress acting on the crack tip. Thus, it is postulated that several processes were involved in increasing the delamination toughness. Debonding of the fiber/matrix interface allowed elastic stretching of the flock fibers. This created fiber bridging of the crack by the flock fiber lines providing crack closure forces until fiber pullout occurred.

5. A simple micromechanical model has been proposed to describe the behavior of z-direction flock fibers in a double-cantilever-beam (Mode I) crack opening, which was observed in the load deflection curves.
6. Overall, z-directional reinforced-with-flock laminates with thermoplastic nylon flock fiber showed the most significant improvement in the delamination strength and fracture toughness.

ACKNOWLEDGEMENT

This research was supported by the National Textile Center Grants F97-D01 and F98-D08. The authors wish to thank Dr. Armand F. Lewis for his encouragement and suggestions during the course of this research work.

REFERENCES

- Cox, Brian, et al 1997. "Delamination Fracture in the Presence of Through-Thickness Reinforcement", *Australian Composite Structures Society*, pp 1-159 + 1-177.
- Gillespie, Jr., J. W. 1991. "Damage Tolerance of Composite Structures", *Journal of Offshore Mechanics and Arctic Engineering*, 247-252.
- Jain, L. K. and R. C. Wetherhold. 1992. "Effect of Extensibility on the Fracture Toughness of Short Fiber or Brittle Matrix Composites, *Applied Mechanics Review*, vol 45, no 8.
- Jain, L. K. and Yiu-Wing Mai. 1994. "On the Effect of Stitching on Mode I Delamination Toughness of Laminated Composites", *Composite Science and Technology*, **51**, 331-345.
- Lewis, A. F. and R. Saxon. 1973. "Epoxy-Resin Adhesives", *Marcel Dekker Inc. New York*
- Morley, J. W. 1987. "High-Performance Fiber Composites", New York; *Academic Press*.
- Reedy Jr., E. D. et al. 1997. "Modeling the Initiation and Growth of Delaminations in Composite Structures", *Journal of Composite Materials*, **31** no 8.
- Takeda, Nobuo. et al 1999. "Effect of Toughened Interlaminar Layers on Fatigue Damage Progress in Quasi Isotropic CRPF Laminates", *International Journal of Fatigue*, **21** 235-242.
- Tanzawa, Yasuyo. et al 1999. "Interlaminar Fracture Toughness of 3-D Orthogonal Interlocked Fabric Composites", *Composite Science and Technology*, **59** 1261-1270.

BIOGRAPHY

Yong K. Kim, a Professor of Textile Sciences at University of Massachusetts Dartmouth, joined the faculty in 1981. He received a Ph.D. in Fiber and Polymer Science from NC State and BS and MS in Textile Engineering from Seoul National University. Yong's research interests include textile process design, mechanics of fibrous structures and micro/nano composite materials.

ANALYSIS ON TEXTILE COMPOSITE CELLULAR STRUCTURES FOR SPECIFIC IMPACT DEFLECTION

Xincai Tan, Xiaogang Chen
Department of Textiles
UMIST, P.O. BOX 88, Sackville Street
Manchester, UK M60 1QD
Email: Xiaogang.Chen@umist.ac.uk

KEYWORDS

Cellular structure, deformation, energy absorption, finite element analysis, textile composite.

ABSTRACT

Cellular structures have been proven useful for energy absorption applications. In order to design, manufacture and apply textile composite cellular structures, it is necessary to understand the mechanisms of energy absorption for these structures. In this paper, the finite element analysis is carried out for energy absorption and deformation in the textile composite cellular structures. Four sets of cellular structures with different cell wall lengths are modelled to evaluate their influence on energy absorption and deformation. The predicted results show that the deformation distributes significantly all over the whole cellular structure during quasi-static impact. For a given dimension of a hexagonal cellular structure with different cell wall lengths, under the same impact deformation, a short cell wall length leads to higher energy absorption capability, although its relative density is correspondingly larger.

INTRODUCTION

Cellular structures are widely used for energy absorption because they are highly compressible, even at very large strains. Man-made polymer, metal, paper, plastics and ceramic cellular structures are now available as standard products (Corden and Bitzer 1987, Gibson and Ashby 1997, Sato 1994), and they are used in a variety of engineering applications, from cheap doors to advanced aerospace components. The honeycomb is formed by expansion and corrugated methods (Corden 1987, Bitzer 1997).

Considerable effort has been made over the past decades on the investigation of cellular structures and composite materials for energy absorption and crash worthiness. Danver *et al.* (1999) traced the manufacturing techniques of honeycomb cores and found that honeycomb cores were not only developed and implemented in aerospace industry, but also migrated to other industries, such as automotive. Honeycomb core reduces the weight of a vehicle yet provides the same safety features required by the public and governmental agencies. A method for the manufacture of honeycomb ceramics with a structure, which yields negative Poisson ratio has been reported (Huang and Blackburn, 2001). Ceramics with cellular or honeycomb structures have applications as substrates in catalytic converters for the emissions from internal combustion

engines. The mechanisms governing in-plane crushing of hexagonal aluminium honeycombs have been investigated with finite size honeycomb specimens crushed quasi-statically between parallel rigid surfaces (Papka and Kyriakides, 1994, 1998). The force-displacement response is initially stiff and elastic but this is terminated by a limited load instability. Finite element (FE) simulations (Gibson and Andrews, 2001) were used to model the steady state, and found that the creep rates of the regular hexagonal honeycomb were lower than other cellular solids.

Composite materials can provide not only a multi-functional multi-layered integrated structure but also low cost manufacturing. In large scale manufacturing of thick composites and integral armour parts of composite armoured vehicle, two low cost manufacturing approaches have been considered that hold promise to produce large size parts – Resin Transfer Moulding and Vacuum Assisted Resin Infusion Moulding (Kelkar and Vaidya 1999). US Army helicopters have been designed (Farley 1987) to meet stringent crash resistance requirements, using energy absorbing sub-floor composite beams which have significantly more energy-absorption capability than comparable metal designs.

Composites with 3D woven, braided or knitted textile composite reinforcements offer a high potential for the cost-effective manufacturing of structures featuring excellent damage tolerance or energy absorption capability. As reviewed by Bibb and Hogg (1996), for different forms of textiles used for composite reinforcement, there is sufficient literature information available to indicate that control of fibre organization by the use of textiles may be an effective method of optimising composite properties for specific end use properties. The cellular composite materials have been fabricated (Small and Ansell 1987) from epoxy resin-impregnated woven glass fabric sheets convoluted into a regular three-dimensional cellular structure. Specific energy absorption values of $> 100 \text{ kJ kg}^{-1}$ have been achieved with a load uniformity approaching unity. Three grid-domed textile composites fabricated from knitted fabrics were studied under quasi-static compression and impact condition (Yu *et al.* 2000). The grid-domed composites exhibiting plastic collapses showed high levels of optimum specific energy-absorbing capacity at higher pressure levels than those of polyester and polyethylene foams. Arendts *et al.* (1993) reported that composites involving the authors' new 3D weavings with minimum fibre crimp could compete with tape-based laminates as far as stiffness and strength were concerned. Using knitting makes it possible to produce complex shaped structures and

to manufacture composites having superior through-penetration resistance. Static crushing tests have been conducted (Chiu *et al.* 1997) on 3-D carbon/epoxy and kevlar/epoxy braided composite square tubes and the results showed that the specific energy absorption capability of these tubes was higher than Kevlar braided composite tube. Textile composite panels reinforced with integrally woven 3-D fabrics have been investigated at UMIST (Zic *et al.* 1990) and their mechanical properties were found to be comparable to those of a softwood, an aluminium alloy, and even a steel.

Most recently, design, analysis and manufacture of textile composite cellular structure (TCCS) have been carried out at UMIST (Chen and Tan 2003). 2D finite element analysis has been performed to compare quasi-static impact results of textile composite cellular structures with those of aluminium alloy ones which are extensively applied in various engineering fields. The predicted results showed that the textile composite cellular structures are of easier deformation and higher energy absorption than those of Al alloy, but they are strong enough to be applied in some load/impact cases.

In order to further understand TCCSs' energy absorbing mechanism, it is necessary to study the distributions of deformation and energy absorption in impacting TCCS. In this paper, after creation of modelled TCCS geometries by an in-house computer package, three-dimensional (3D) FE modelling is carried out to investigate the influences of cell wall length on energy absorption and displacement distributions on the surfaces of TCCSs during quasi-static impact.

GEOMETRY CREATION

It has been realised that creation of geometry of TCCSs in commercial FE package before FE analysis demands not only considerable computing skill from the user, but also lengthy time. In addition, an FE package such as MSC.MARC Mentat only supports single block (4 points for 2D and 8 points for 3D) for each part of geometry/element input. It takes a long time to create a cellular structure with many cell wall blocks, because each cell wall is considered as a single block and has to be treated individually.

In order to conduct FE analysis efficiently, an in-house package, used as a pre-processor, has been developed at UMIST, whose interface is shown in Fig.1. After specification of the geometric parameters, such as wall length and wall thickness, an IGES file of the cellular structure geometry are generated. This standard IGES file can be exported and read by commercial CAD and FEM packages.

FINITE ELEMENT MODELLING

The quasi-static impacts of TCCSs were simulated with 3D FE models using MSC.Marc Mentat 2001. The FE simulations were carried out as isothermal, linear elastic, and double symmetric problem in which only a fourth of the structure was modelled. The dimensions for all cellular

structures were the same, $20 \times 30 \times 50$ mm in thickness \times width \times length, and an example is shown in Fig.1a. For all impact cases, a foreign object was assumed to impact on the centre of cellular structure, and the impacted area was 12×20 mm. Boundary conditions were used to enforce symmetry in X-Z plane and Y-Z plane. In the simulations, four sets of different original cell wall lengths were chosen, which are (a) 2mm, (b) 3 mm, (c) 4 mm, and (d) 6 mm. The original free wall length and the original bonded wall length were always kept the same for each set. The original free cell wall thickness and the original bonded cell wall thickness were initially 0.5 mm and 0.6 mm, respectively. The original 3D FE meshes for TCCSs with different cell wall lengths are shown in the second column of Fig.2. The final deformation of the centre point on the top surface of a structure after impact was given a certain magnitude, 5mm.

The material of the modelled TCCSs was chosen from the previous work (Zic *et al.* 1990), the fibres were R136/2-tex EC9 glass; and the fabrics were converted into composite cellular structures by impregnating by mixture of resins, Araldite LY 1927 GB with HY 1927 GB. The experimental measured Young's modulus for the textile composite material was 13.2 GPa and the Poisson ratio was given as 0.3.

The total strain energy absorbed by each specimen was calculated as the area beneath the load-deflection trace. The strain energy density was calculated as the strain energy absorbed per unit volume of impacted structure. The deformation was the resultant value of displacements in X, Y and Z directions. The predicted local result data (such as the strain energy density, the deformation) were collected from the nodes of the top and bottom horizontal outer surfaces of the structures.

RESULTS AND DISCUSSIONS

Deformation

The deformed meshes after impact for textile composite cellular structures with different cell wall lengths are shown in the last column of Fig.2. Within the impacted area, the horizontal cell walls of the top surface appear serious deflections after impact, whereas the declined cell walls bend almost in the same way.

The predicted deformation distributions on top surfaces for TCCSs with various cell wall lengths are shown in the second column of Fig.3. The deformations for all TCCSs are significantly seen all over the fabric. For outer horizontal surface of each unit cell within the impacted area, the farther the distance from the central lines (X-Z plane and Y-Z plane in Fig.1a) outside the impacted area, the smaller the deformation. In total, the deformations along width (or X) direction are larger than those along length (or Y) direction. The main reason for this is because the material in width direction is a continue wall sheet which deforms steadily and continuously in the same cell wall, whereas the material in length direction is discontinuous where deformation has to be transferred from one cell to another cell. The smaller the cell wall length, the

smaller the deformation outside the impacted area, although these deformations are all pronounced.

The predicted deformation distributions on bottom surfaces for cellular structures with various cell wall lengths can be seen in the last column of Fig.3. The deformations along the width direction for a given horizontal cell wall are almost constant, whereas along the length direction, the deformations appear larger and larger outwards from the centre line. This means that during impact due to the effect of other cells' deformation, the bottom horizontal cell walls slide in length direction but those in central line were kept stationary.

Energy absorption

The predicted total strain energy absorbed and relative density versus cell wall length is presented in Fig.4. Under the present modelled process conditions, the TCCS with smaller cell wall length absorbs higher strain energy during impact. This implies that if the cell wall is given and heavy load is applied, short cell wall length has to be employed. It is noted that the smaller the cell wall length, the higher the relative density, and the more the material and the higher the structure weight will be.

The predicted strain energy density distributions on top surface for cellular structures with various cell wall lengths are plotted in the second column in Fig.5. The strain energy densities within the impacted area are much higher than those outside the impacted area. Similar to deformations in the second column of Fig.3, for outer horizontal surface of each unit cell within the impacted area, the farther the distance from the central line, the smaller the strain energy density; in total, the strain energy density along width (or X) direction are larger than those along length (or Y) direction. In fact, the strain energy densities here are proportional to the deformations.

The predicted strain energy density distributions on bottom surfaces for TCCSs with various cell wall lengths are seen in the last column of Fig.5. For a given horizontal cell wall on the bottom, contrary to the top, along the length direction, the strain energy densities are larger and larger towards the centre line of the cell wall. Along the width direction, the strain energy densities vary more gently than those on the top surface, although this fact is mainly seen for the impacted cell columns (Z direction). This is because the inclined cell walls of the bottom unit cells mainly support the impact load.

CONCLUSIONS

- (1) 3D finite element modelling has been carried out to investigate into energy absorption and deformation in impacting textile composite cellular structures.
- (2) The deformation appears for all unit cells of whole textile composite cellular structure during quasi-static impact. On the top surface, deformations in the impacted areas develop larger than those outside the impacted areas; but on the bottom surface in the length direction, the farther the distance from the central line, the larger the deformation.

- (3) The strain energy densities mainly appear in the impacted region. Under the present modelled process conditions, a cellular structure with a smaller cell wall length is of higher energy absorption capability, although its relative density is correspondingly larger.

REFERENCES

- Arendts, F.J., Drechsler, K. and Brandt, J. (1993) Advanced textile structural composites--status and outlook, *Advanced Composites '93*, Wollongong, Australia, 15-19 Feb. 1993, pp. 409-416.
- Bibo, G.A. and Hogg, P.J. (1996) Role of reinforcement architecture on impact damage mechanisms and post-impact compression behaviour, *Journal of Materials Science*, Vol. 31, No. 5, pp. 1115-1137.
- Bitzer, T. (1997). *Cellular structure technology: materials, design, manufacturing, applications and testing*, Chapman & Hall, London.
- Chen, X. and Tan, X. (2003) Comparisons of cellular structures between textile composite and Al alloy, *Computational Engineering in Systems Applications (CESA 2003)*, July 2003, Lille, France.
- Chiu, C.H., Lu, C.K. and Wu, C.M. (1997) Crushing characteristics of 3-D braided composite square tubes, *Journal of Composite Materials*, Vol.31, No.22, pp.2309-2327.
- Corden, J. (1987) *Honeycomb Structure*, ASM International, *Engineered Materials Handbook*. Vol.1: Composites, pp.721-728.
- Corden, J.L. and Bitzer, T.N. (1987) *Honeycomb materials and applications*, *Advanced Materials Technology '87*, Anaheim, California, USA, 6-9 Apr. 1987. pp. 68-79.
- Farley, G.L. (1987) Energy absorption in composite materials for crashworthy structures, 6th Int'l Conf. on Composite Materials and 2nd European Conference on Composite Materials (ICCM & ECCM). London, UK, 20-24 July 1987, Elsevier, Vol. 3, pp. 3.57-3.66.
- Danver, D., Ciobanu, S. and Niemerski, M. (1999) Honeycomb core utilization in the automotive industry, *SAMPE Advanced Composites Conference Proceedings (USA)*, 27-28 Sept. 1999, pp. 231-243.
- Gibson, L.J, Ashby, M.F. (1997). *Cellular solids: structure and properties*, Second Edition, Cambridge University Press, Cambridge, UK.
- Gibson, L.J, Andrews, E.W. (2001). The role of cellular structure in creep of two-dimensional cellular solids, *Materials Science and Engineering A*, Vol.303, No.1-2, pp.120-126.
- Huang, X, Blackburn, S. (2001). Developing a new processing route to manufacture honeycomb ceramics with negative Poisson's ratio, *Proceedings of the 7th Conference & Exhibition of the European Ceramic Society (Part VII)*, Brugge, Belgium, 9-13 Sept. 2001, *Key Engineering Materials*, Vol. 206-213, No. 1, pp. 201-204, 2002.
- Kelkar, A.D. and Vaidya, U. (1999) Low cost manufacturing of composite integral armor using resin transfer and resin infusion molding processes, *Proceedings the 5th Int'l Conf on Flow Processes in Composite Materials*, Plymouth, UK, 12-14 July 1999, pp. 321-328.

- Papka, S, Kyriakides, S. (1994). In-plane compressive response and crushing of cellular structure, Journal of the Mechanics and Physics of Solids, Vol.42, No.10, pp.1499-1532.
- Papka, SD; Kyriakides, S. (1998). Experiments and full-scale numerical simulations of in-plane crushing of a honeycomb, Acta Materialia, Vol.46, No.8, pp.2765-2776.
- Sato, T. (1994) Summary of basic honeycomb structural material--kinds and functions, Kino Zairyo (Function and Materials) (Japan), Vol. 14, No. 7, pp. 52-65.
- Small, G.D. and Ansell, M.P. (1987) The Energy-Absorbing Properties of a Novel Cellular Structure, Journal of Materials Science, Vol. 22, No. 8, pp. 2717-2722.
- Yu, T.X., Tao, X.M. and Xue, P. (2000) The energy-absorbing capacity of grid-domed textile composites, Composites Science and Technology, Vol. 60, No. 5, pp.785-800.
- Zic, I., Ansell, M.P., Newton, A., Price, R.W. (1990). Mechanical properties of composite panels reinforced with integrally woven 3-D fabrics, Journal of the Textile Institute, Vol.81, pp.461-479.

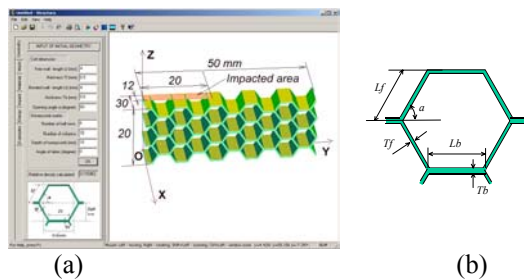


Fig.1 (a) Geometry package (b) a unit cell

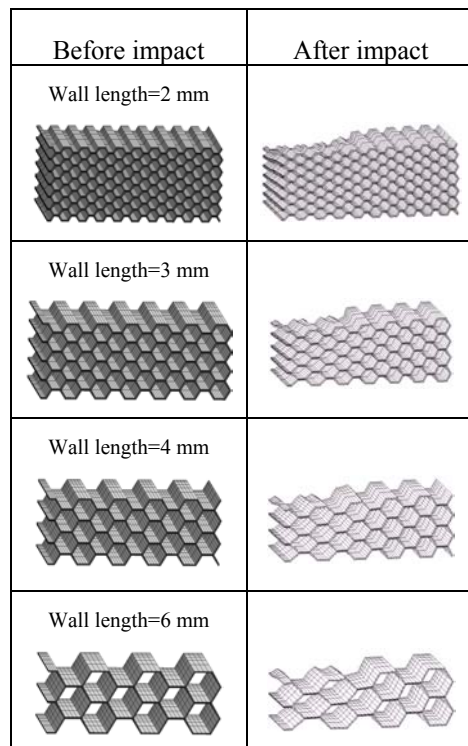


Fig. 2 3D finite element meshes before and after impact

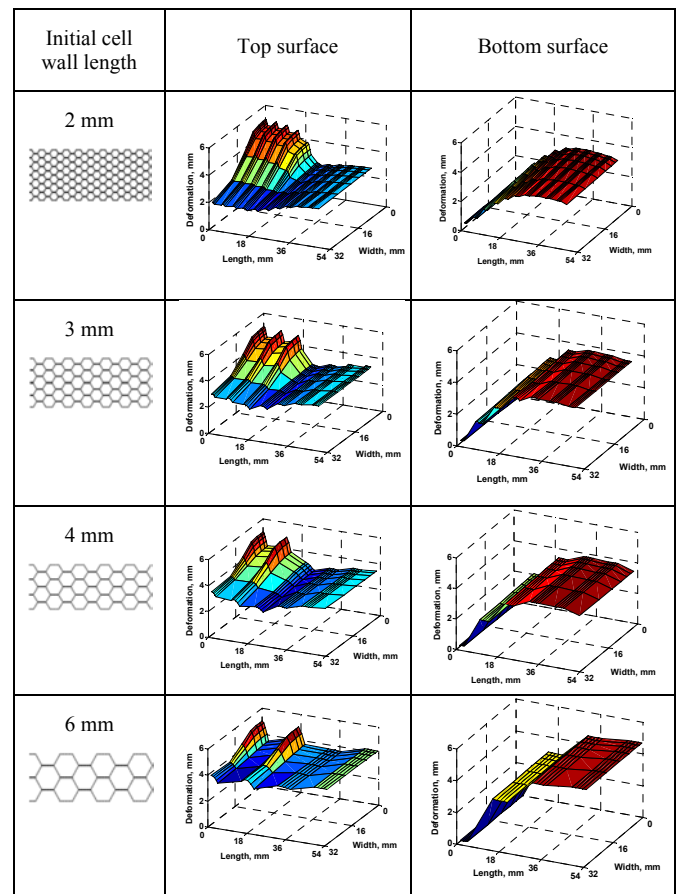


Fig.3 Predicted deformation distributions

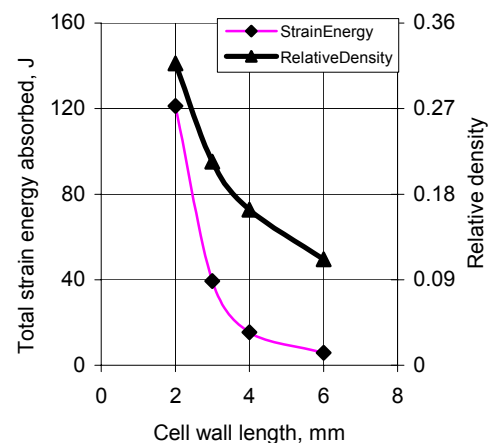


Fig.4 Total strain energy absorbed and relative density versus cell wall length

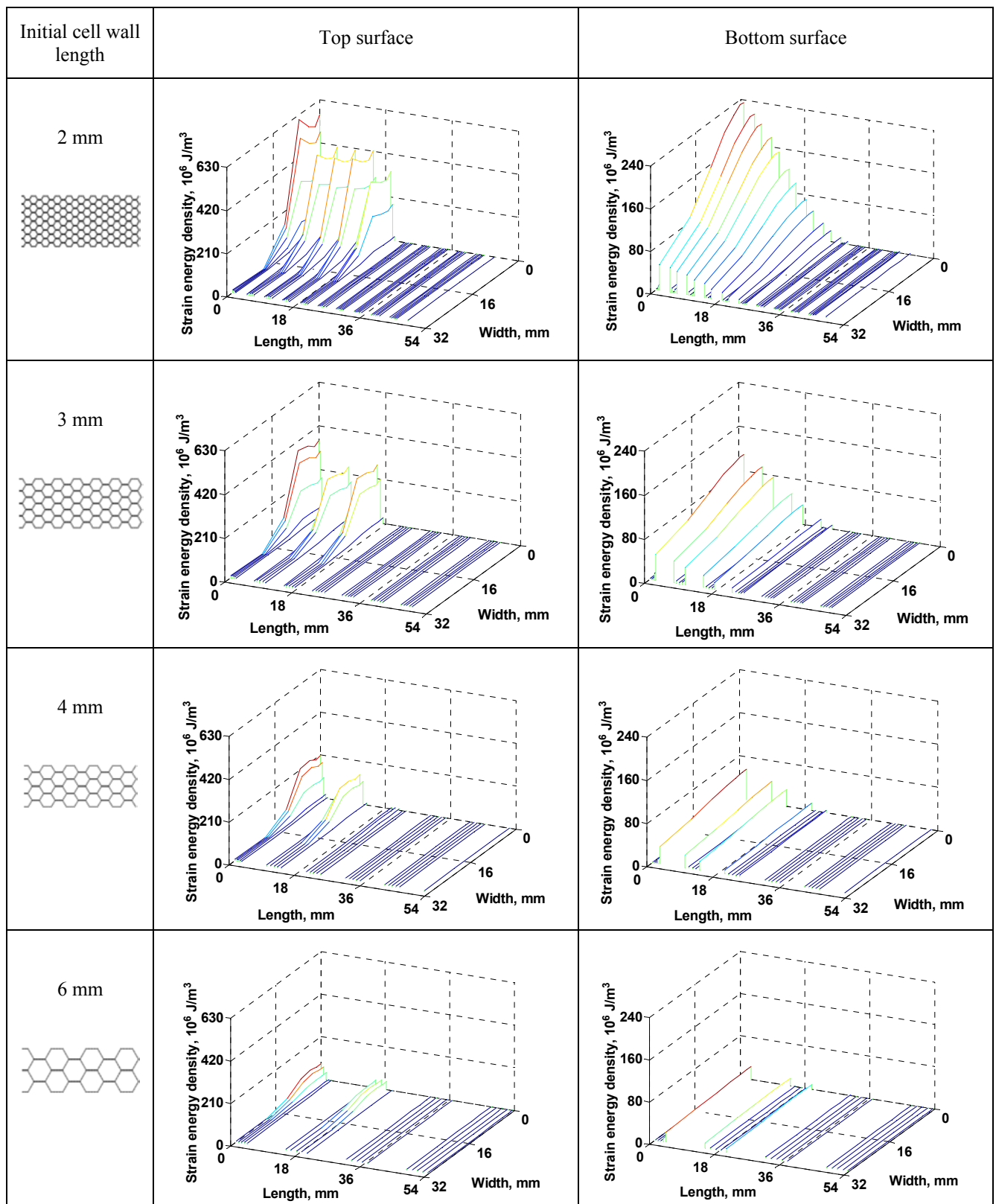


Fig.5 Predicted strain energy density distributions on top and bottom surfaces

Virtual Textiles I

Exploring the Nature of the Air Drawing of Polymer in the Melt Blowing Nonwoven Process via Numerical Simulations

Ting Chen and Xiubao Huang
College of Textiles
Donghua University
Shanghai, 200051
China
E-mail: chen@ensait.fr

KEYWORDS

Melt Blowing, Microfiber, Air Drawing, Flow Field, Numerical Simulation

ABSTRACT

The air drawing model of polymer and the air jet flow field model in the melt blowing process are founded. The flow field model is solved using the finite difference method. The numerical computation results of distributions of x -component of air velocity and air temperature along x -axis and y -axis coincide with the experimental data. The air drawing model of polymer is solved with the aid of the distributions of air velocity and air temperature. The model prediction of the fiber diameter tallies with the experimental data well. The results show great perspective of this research in the field of computer assisted design of melt blowing technology and equipment.

INTRODUCTION

Melt blowing is used commercially as a one-step process for converting polymer resin directly into a nonwoven mat of fibers. In the melt blowing process, a molten stream of polymer is extruded from the die and rapidly attenuated into microfibers by the air jet with high velocity and high temperature. Researchers have been devoting to the study of the air drawing model of polymer (Uyttendaele and Shambaugh 1990; Harpham and Shambaugh 1996; Harpham and Shambaugh 1997; Bansal and Shambaugh 1998). However, the reported air drawing models are all

based on the experimental results of the air jet flow field. In this paper an air drawing model of polymer will be founded based on the numerical simulation results of the air jet flow field. The final fiber diameter then can be predicted with the aid of the air drawing model.

THE AIR DRAWING MODEL OF POLYMER

The air drawing model of polymer consists of continuity equation, momentum equation, energy equation and constitutive equation. The surrounding air conditions (velocity and temperature) will be considered as given functions of axial position.

Continuity equation:

$$G = \frac{\pi}{4} D^2 u \rho \quad (1)$$

where

G = polymer flow rate

D = fiber diameter

u = fiber velocity

ρ = polymer density

As the polymer density varies with polymer temperature, the correlation given by Zieiminski and Spruiell (Zieiminski and Spruiell 1986) is introduced:

$$\rho = \frac{1}{1.145 + 0.000903 \cdot \theta} \quad (2)$$

where

θ = polymer temperature

Momentum equation:

$$\frac{dF_r}{dx} = G \frac{du}{dx} + \frac{1}{2} \pi D C_f \rho_a (u_a - u)^2 - \frac{\pi}{4} D^2 \rho g \quad (3)$$

where

F_r = rheological force

ρ_a = air density

x = axial position

u_a = x -component of air velocity

C_f = air drawing coefficient

g = gravitational acceleration

The rheological force is

$$F_r = \frac{\pi}{4} D^2 (\tau_{xx} - \tau_{yy}) \quad (4)$$

where

τ_{xx} = axial tensile stress of polymer

τ_{yy} = transversal tensile stress of polymer

The air drawing coefficient C_f was given by Matsui (Matsui 1976) with the following correlation

$$C_f = \beta \cdot Re^{-n} \quad (5)$$

where

β, n = constants of Matsui's correlation

Re = Reynolds number

According to Majumdar and Shambaugh (Majumdar and Shambaugh 1990), the assumed values of β and n are 0.78 and 0.61, respectively.

Energy equation:

$$\frac{d\theta}{dx} = -\frac{\pi D h_p (\theta - \theta_a)}{G C_p} \quad (6)$$

where

C_p = specific heat capacity at constant pressure of polymer

h_p = heat transfer coefficient

θ_a = air temperature

Implicit in Equation (6) is no conductive resistance in the radial direction and no viscous dissipation.

Like the polymer density, the specific heat capacity at constant pressure of polymer varies with polymer temperature. The correlation given by Zieiminski and Spruiell (Zieiminski and Spruiell 1986) is introduced:

$$C_p = 0.3669 + 0.00242 \cdot \theta \quad (7)$$

Constitutive equation:

As is well known, polymer melt is a kind of non-Newtonian fluid. The constitutive equation of power-law fluid is introduced in our model:

$$\tau_{xx} = 2\eta \left(\frac{du}{dx} \right)^m \quad (8)$$

$$\tau_{yy} = -\eta \left(\frac{du}{dx} \right)^m \quad (9)$$

where

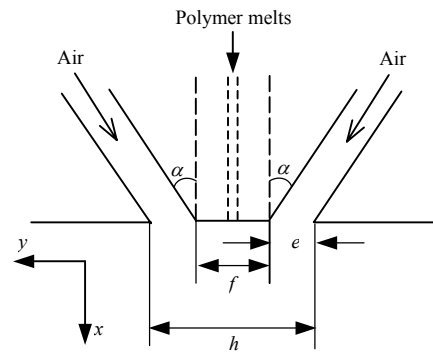
η = shear viscosity

m = power-law exponent

The 'freezing point' is defined as the boundary condition where the rheological force is considered to be the sum of the cumulative gravitational and air drawing forces acting upon the frozen part of the fiber. Beyond the 'freezing point', the fiber diameter remains constant until the fiber is laid down on a collection screen.

NUMERICAL SIMULATION OF THE AIR JET FLOW FIELD

As the fiber diameter is strongly affected by the air jet flow field of the dual slot die shown in Figures 1, researchers have paid great attention to the simulations of the flow field, especially Harpham and Shambaugh (Harpham and Shambaugh 1996; Harpham and Shambaugh 1997) have made valuable contributions to the experimental study. In the previous paper (Chen and Huang 2001), a theoretical model of the flow field was established by substituting the jets with the potential flows. Considering the complication of the flow field that is formed by the colliding and mixing of the two air jets with high velocity and high temperature, it is necessary to simulate the flow field numerically.



Figures 1: Dual Slot Die of the Melt Blowing Process

The Air Jet Flow Field Model

The flow of two jets is assumed to be steady, viscous flow. As $k-\varepsilon$ model is the most widely used turbulence model in engineering computations and will make the simulated results more accurate in the case of high Reynolds numbers, it is taken as the preferred turbulence model.

The air jet flow field model consists of continuity equation, momentum equation, energy equation, turbulent kinetic energy equation and turbulent dissipation rate equation. According to reference (Huai and Li 1994), values of constants of k - ε model: $C_\mu = 0.09$, $C_{\varepsilon 1} = 1.44$, $C_{\varepsilon 2} = 1.92$, where C_μ , $C_{\varepsilon 1}$, and $C_{\varepsilon 2}$ are constants of k - ε model. Turbulent Prandtl numbers are as follows: $\sigma_t = 0.85$, $\sigma_k = 1.0$, $\sigma_\varepsilon = 1.3$, where σ_t , σ_k , and σ_ε are Prandtl numbers of turbulence, of turbulent kinetic energy and of turbulent dissipation rate.

Boundary conditions:

As the flow field is symmetrical along the system centerline, half of the plane is chosen as the computation area. The upstream section is on the lower edge of the die head. The downstream section is considered to be far enough from the front edge. The boundaries far enough from the system centerline are taken as the outer boundaries in y direction. Then the boundary conditions of this model can be obtained:

(1) The conditions of upstream sections without inlet are:

$$u_a = 0 \quad v_a = 0 \quad \frac{\partial \theta_a}{\partial x} = 0 \quad k_a = 0 \quad \varepsilon_a = 0$$

where

u_a = x -component of air velocity

v_a = y -component of air velocity

θ_a = air temperature

k_a = turbulent kinetic energy of air

ε_a = turbulent dissipation rate of of air

The conditions of upstream sections with inlet (in the slot) are:

$$u_a = u_{j0} \quad v_a = v_{j0} \quad \theta_a = \theta_{j0}$$

$$k_a = 0.06(u_{j0}^2 + v_{j0}^2) \quad \varepsilon_a = 0.06 \frac{u_{j0}^3 + v_{j0}^3}{e}$$

where

u_{j0} = x -component of initial air velocity

v_{j0} = y -component of initial air velocity

θ_{j0} = initial air temperature

The downstream section:

$$\frac{\partial u_a}{\partial x} = \frac{\partial \theta_a}{\partial x} = \frac{\partial k_a}{\partial x} = \frac{\partial \varepsilon_a}{\partial x} = 0 \quad v_a = 0$$

Centerline condition:

$$\frac{\partial u_a}{\partial y} = \frac{\partial \theta_a}{\partial y} = \frac{\partial k_a}{\partial y} = \frac{\partial \varepsilon_a}{\partial y} = 0 \quad v_a = 0$$

Outer boundaries:

$$u_a = k_a = \varepsilon_a = 0 \quad \theta_a = \theta_{am} \quad \frac{\partial v_a}{\partial y} = 0$$

where

θ_{am} = ambient temperature

Numerical Methods for the Air Jet Flow Field Model

The air jet flow field model is solved using the finite difference method as follows: (a) using SIMPLE algorithm (Shaw 1992) to solve velocity-pressure coupling; (b) introducing staggered grid to avoid tooth-like distributions of velocity and pressure; (c) adopting power-law scheme as the preferred difference scheme; (d) viscosity and density on the interface are arithmetic mean of neighboring nodes; (e) using TDMA method to solve difference equations.

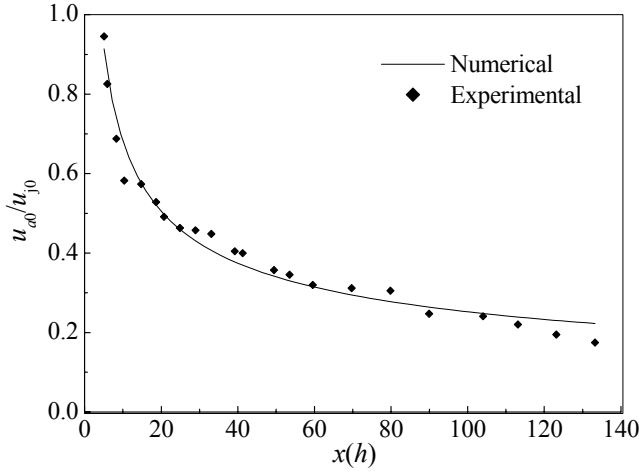
The computational domain is rectangular where the coordinate origin is at the center of the die head. The x direction is along the spinneret axis. The lengths of x -direction and y -direction of the computational domain are 1400 mm and 400 mm respectively. There are 1000 grids in x direction and 800 grids in y direction.

Comparison of Computed Results with Experimental Data

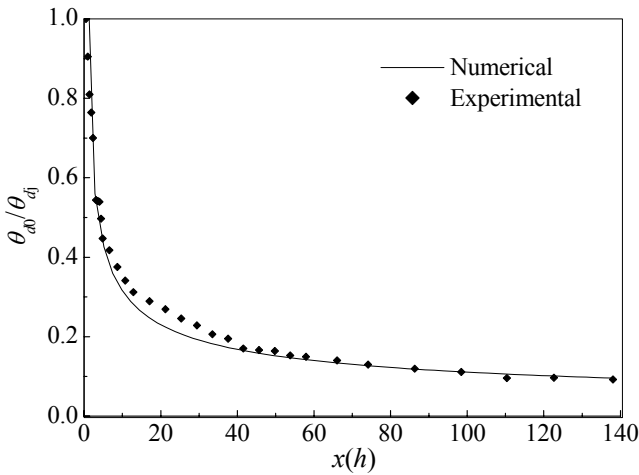
Experiments are done on the flow field of the melt blowing die shown in Figures 1. The air velocity and air temperature are measured by the Laser Doppler Velocimeter and the Hot Wire Anemometer, respectively. The die width $h = 5$ mm. The die length $l = 50$ mm. The slot width $e = 1$ mm. The angle between the air slot and the spinneret axis $\alpha = 30^\circ$. The head width $f = 3$ mm. The initial air velocity is 100 m/s. The initial air temperature is 130 °C. The measured domain is the rectangle below the die with the size in x direction of 1400 mm and in y direction of 400 mm. Air velocities and air temperatures of over 600 positions are measured.

The distributions of the centerline x -component of air velocity along x -axis and air temperature along x -axis are shown in Figures 2 and 3 respectively. Dots in the figures are the experimental data. In the figures θ_d is the difference between the air temperature and the ambient temperature and θ_{d0} is the difference between the air temperature along the centerline (the x -axis) and the ambient temperature. And $x(h)$ is the nondimensionalized

axial position, $x(h) = (x/h) (\rho_\infty/\rho_0)^{1/2}$, where ρ_∞ is the air density at ambient conditions and ρ_0 is the air density along centerline downstream from the die. As can be concluded, the theoretical results obtained with the model tally with the experimental data well.



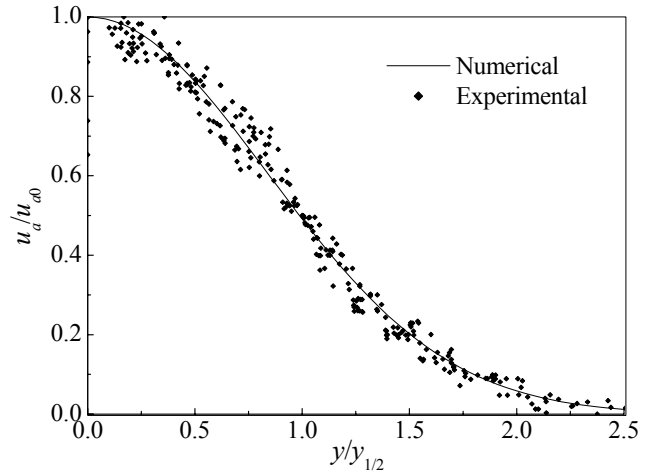
Figures 2: Distribution of centerline x-component of air velocity along x-axis



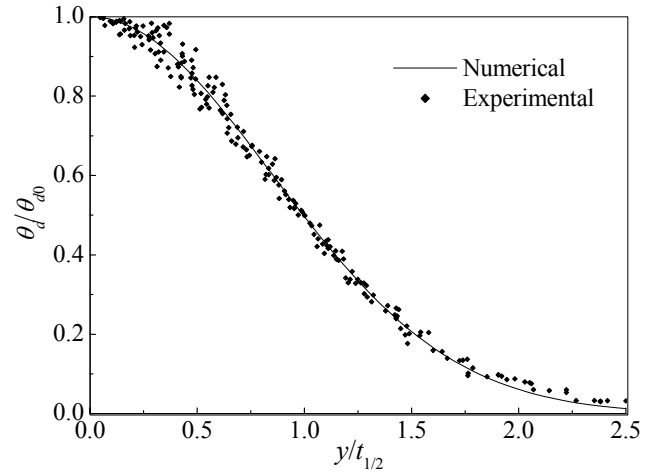
Figures 3: Distribution of centerline air temperature along x-axis

The distributions of the x-component of air velocity along y-axis and air temperature along y-axis are shown in Figures 4 and 5 respectively. Dots in the figures are the experimental data. In the figures $y_{1/2}$ and $t_{1/2}$ are the value of the coordinate point along y direction where the air velocity component in x direction drops to half of its maximum value and where the air temperature drops to half of its maximum value, respectively. As can be concluded, the theoretical results obtained with the model are coincident with the experimental data well too. The simulation profiles are the average of profiles for positions

of $x = 14.5, 75$ and 155 mm since they are very close to each other.



Figures 4: Distribution of centerline x-component of air velocity along y-axis



Figures 5: Distribution of centerline air temperature along y-axis

NUMERICAL SIMULATION OF THE AIR DRAWING OF POLYMER

With the aid of numerical simulations of the air jet flow field, how the x-component of air velocity u_a and air temperature θ_a change with the axial position x can be found out. Then the air drawing model of polymer can be solved using a fourth order Runge-Kutta method. Experiments are done on the melt blowing equipment at Donghua University. The initial air velocity is 168 m/s.

The initial air temperature is 310 °C. The polymer used in the experimental runs is 36 MFR (melt flow rate) YMC9747 polypropylene. The power-law exponent $m =$

0.78. The polymer mass flow rate is 0.035 g/s. The initial polymer temperature is 260 °C. The image analysis method is employed to measure the fiber diameter. Diameters of two hundred fibers chosen randomly on the nonwoven samples are averaged to obtain the mean fiber diameter. The measured fiber diameter is 11.600 μ m and the computed fiber diameter is 10.499 μ m, which proves that the numerical simulation presented in this paper is effective.

CONCLUSION

The air drawing model of polymer and the air jet flow field model are founded. The flow field model is solved numerically with the finite difference method. The computation results of distributions of x -component of air velocity along x -axis and y -axis and distributions of air temperature along x -axis and y -axis coincide with the experimental data. The distributions of air velocity and air temperature are introduced into the air drawing model of polymer. The model prediction of the fiber diameter tallies with the experimental data well. The results show great perspective of this research in the field of computer assisted design of melt blowing technology and equipment.

REFERENCES

- Uyttendaele, M. A. J. and R. L. Shambaugh. 1990. "Melt Blowing: General Equation Development and Experimental Verification." *AIChE Journal* 36, No.2, 175-186.
- Harpham, A. S. and R. L. Shambaugh. 1996. "Flow Field of Practical Dual Rectangular Jets." *Industrial and Engineering Chemistry Research* 35, No.10, 3776-3779.
- Harpham, A. S. and R. L. Shambaugh. 1997. "Velocity and Temperature Fields of Dual Rectangular Jets." *Industrial and Engineering Chemistry Research* 36, No.11, 3937-3943.
- Bansal, V. and R. L. Shambaugh. 1998. "On-Line Determination of Diameter and Temperature during Melt Blowing of Polypropylene." *Industrial and Engineering Chemistry Research* 37, No.5, 1799-1806.
- Zieiminski, K. F. and J. E. Spruiell. 1986. "A Mathematical Model of Crystalline Fiber-Forming Polymers." *Synthetic Fibers*, No.4, 31-38.
- Matsui, M. 1976. "Air Drag on A Continuous Filament in Melt Spinning." *Transactions of the Society of Rheology* 20, No.3, 465-473.
- Majumdar, B. and R. L. Shambaugh. 1990. "Air Drag on Filaments in the Melt Blowing Process." *Journal of Rheology* 34, No.4, 591-594.
- Chen T. and X. Huang. 2001. "Study on the Flow Field of the Air Jet From Dual Slot Die in the Melt Blowing Process." *Polymer Engineering and Science* 41, No.6, 971-976.
- Huai, W. and W. Li. 1994. "Hybrid Finite Analytic Solutions of Coanda Regions of Annular and Twin Parallel Jets." *Journal of Hydrodynamics Series A* 9, No. 1, 112-119.
- Shaw, C. T. 1992. *Using computational fluid dynamics*. Prentice Hall, New York.

AUTHOR BIOGRAPHY

TING CHEN was born in Shandong, China and went to China Textile University, where he obtained his doctoral degree in 2002. His doctoral thesis mainly concerns the numerical simulation of the air drawing of polymer and the air jet flow field in the melt blowing nonwoven process. Now he is doing postdoctoral researches at ENSAIT, which concerns the modeling of the relationship between the structural parameters and functional properties of nonwovens. E-mail: chen@ensait.fr

A SYSTEM FOR THE VIRTUAL DESIGN AND SIMULATION OF GARMENTS

Marzia Fontana

Dipartimento di Ingegneria Industriale
Università di Parma
Parco Area delle Scienze, 181/A
43100 - Parma, Italy
E-mail: marzia.fontana@kaemart.it

Caterina Rizzi

Dipartimento di Ingegneria
Università degli Studi di Bergamo
Via Marconi, 5
24044 - Dalmine (BG), Italy
E-mail: crizzi@unibg.it

Umberto Cugini

Dipartimento di Ingegneria Meccanica
Politecnico di Milano
Via La Masa, 34
20156 - Milano, Italy
E-mail: umberto.cugini@polimi.it

KEYWORDS

CAD, physics-based modelling, simulation, apparel design.

ABSTRACT

A system is presented for the computer aided design and virtual simulation of garments and textiles. Garment models are handled through a multi-level design process including surface modelling tools, manufacturing aspects, and a physics-based cloth description for a final, accurate simulation of garments over mannequins in virtual scenes. Modelling and simulation algorithms are described. The modelling phase includes 3D cloth geometries, 2D pattern definition and geometrical description of textile operations, such as definition of darts and seams, layers, buttons, etc. The physics-based model considered for the simulation phase consists of a discretized representation of garments through structured particle systems embedded in constrained Newtonian dynamics. The system has been validated within European and national projects for virtual garment prototyping. As application, several examples of women's and men's garment models, such as skirts, tunics, jackets, etc. are analysed and simulation results are presented.

1. INTRODUCTION

Automation and CAD systems are, nowadays, increasingly spreading even in a sector such as the clothing industry, typically more anchored to traditional design and manufacturing methodologies and skilled-human labour dependent.

In this sector, in fact, quite well established CAD systems exist for textile printing, 2D pattern drafting and sizing, pattern nesting, marker making, together with CAM tools assisting garment manufacturing processes, e.g. NC devices for fabric cutting and sewing. On the other hand, a CAD approach based on 3D cloth shape design is still at an early stage, although this is particularly demanded by clothing companies to fasten the process of cloth design and make it closer to stylists' traditional way of working by hand-sketching (see Section 2). Beyond 3D geometric modelling tools, more enhanced capabilities such as virtual cloth analysis and 3D simulation are encouraged, to evaluate/improve the quality of garment models before production, reduce the number of intermediate prototypes and, definitively, save time and cost.

Several *geometry-* or *physics-based* methods for modelling cloth have been proposed by the scientific community since the last two decades (IEEE 1996; House and Breen 2000). While the former conceive cloth as a mere geometrical entity, physics-based approaches analyse cloth within a

physical framework, taking into account mechanical properties depending on materials. Geometrical cloth models can be adequate representations for animation purposes in computer graphics. Nevertheless, physics-based cloth models ensure a more accurate shape description, becoming thus suitable tools for a new conception of apparel-oriented CAD systems able to realistically predict cloth's behaviour for virtual prototyping tasks.

The system for virtual cloth design here presented has been developed according to these guidelines. Garment models are analysed as results of a *construction process*, by considering an initial 3D geometrical model, the successive extraction of 2D patterns and the definition of textile operations, such as sewing rules, dart and button insertion, multi-layered fabrics, and material composition (Section 3). The system includes a physics-based cloth model (Section 4) for the static/dynamic simulation of garments in 3D virtual scenes. The data structure and the main modelling and simulation algorithms of the system are presented in Section 5. The system has been experimented and validated in European projects for virtual cloth design and prototyping. Section 6 presents working phases and results for the virtual simulation of male and female garments, presenting several levels of design complexities, e.g., dresses, skirts, jackets. Conclusions and proposals for future work follow in Section 7.

2. OUTLINE ON EXISTING SYSTEMS

In the field of CAD/CAM, several computer-based systems have been developed to assist cloth designers and manufactures. These are essentially based on two different approaches.

A first class of systems provides tools that initially require 2D pattern drawing and, successively, modelling/editing of cloth shapes in 3D space. In this case, 2D pattern shapes are mapped onto virtual mannequins and assembled together along sewing lines. MiraCloth, a well-known cloth modelling system developed at Miralab laboratory, University of Geneve, is based on this idea (Volino and Thalmann 2000). This 2D-to-3D methodology is still the most largely used on the market, with a well-established CAD technology on 2D pattern design through parametric pattern editing, sizing, grading and nesting functions. Well-known industrial packages follow this approach. These are, for instance, *PAD System* from PAD System Technology (www.padsystem.com), the *Optitex* products from Scanvec Garment Systems (<http://www.optitex.com/index.html>), or products from Lectra Systemes (<http://www.lectra.com/en/index.php>) and Gerber Technology (<http://www.gerber-technology.com/gtwww998/library/v19n2/apstd.htm>).

Only a few systems, such as the *Body Garment* package from Investronica Sistemas (see <http://www.investronica-sis.es/>), adopt a 3D-to-2D approach. This is also the approach proposed in the present work. In this case, cloth shapes are first designed in 3D space and, successively, the corresponding 2D profiles of fabric patterns are extracted, through flattening and trimming operations. Other CAD prototypes based on garment definition in 3D space have been proposed mainly by research institutes, such as the three-dimensional apparel-oriented CAD system provided by H. Imaoka and H. Okabe (Okabe et Al. 1992), CAD prototypes developed at Queen's University of Belfast (Hinds and McCartney 1990) and, more recently, the *3D Garment System* package, developed at Hong Kong University of Science and Technology (Leung et Al. 1997) (www.cadcam.ust.hk/research/garment.html), and the system developed by T. Bonte's team at the University of Valenciennes (LAMIH 1999). Figure 1 shows a snapshot of Bonte's system for the design of women's garments.

Examples of commercial packages for cloth animation are *MayaCloth*, plug-in for Maya from Alias-WaveFront (<http://www.aliaswavefront.com/en/products/maya/index.shtml>), *ClothReyes* from Reyes InfoGraphica (<http://www.reyes-infografica.com/cloth.php>) and *SimCloth* from Chaos Group (<http://www.chaosgroup.com/SimCloth.html>), plug-ins for 3D Studio Max, *Stitch1.1* from Digimation (http://www.digimation.com/asp/product.asp?product_id=261), and *TopixCloth*, plug-in for Softimage V3.7 from Topix (<http://www.topix.com/cloth/>), containing a physics-based simulation module. Among these animation systems, however, only *Stitch1.1* includes some basic CAD functionalities: these are, in fact, mainly used for cloth visualization in animated movies.

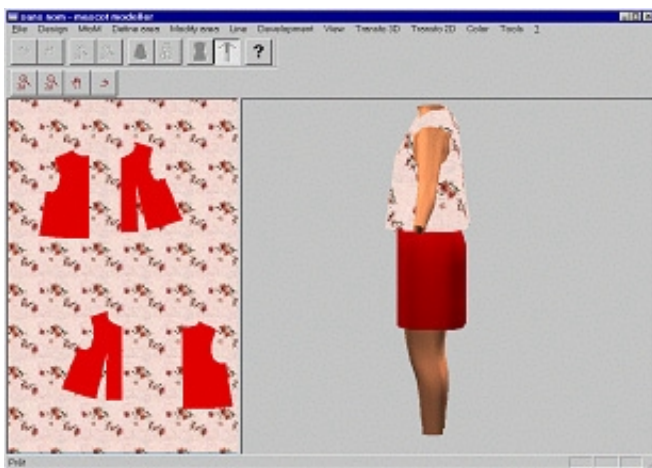


Figure 1. An academic CAD system for garment modelling (courtesy of T. Bonte).

3. GARMENT AS A DESIGN PROCESS

Garments are designed in a large variety of shapes, textile materials, finishings and manufacturing operations. This variety is dictated by several choices due to functional and aesthetic criteria, available materials, available technologies and resources. Despite such differentiation, when analysing garments as “construction processes” we can single out conceptually similar stages that can be considered to define a common computer-assisted garment design methodology.

The way of working of apparel stylists and manufacturers can be, somehow, analysed as a *3D-to-2D-to-3D* process. This starts, in fact, with stylists' creative ideas originating essentially from a 3D shape (a conceptual idea in mind or an already existing garment) from which 2D informations are extracted, such as 2D patterns with corresponding fabric layers (3D-to-2D stage). 2D fabric panels are then assembled and sewn together to get a 3D garment shape as close as possible to the original stylist' idea (2D-to-3D stage). From a procedural point of view, of course, a more detailed sequence of design/manufacturing steps has to be considered:

- (*initial idea*) definition of a reference 3D model;
- (*design*) 2D pattern definition/extraction (2D models);
- (*design*) definition of assembly rules (seams, darts, overlapped layers, buttons, etc.);
- (*design*) definition of materials;
- (*manufacturing*) cutting of 2D fabrics (single layers);
- (*manufacturing*) fabric assembly by layer overlapping;
- (*manufacturing*) one- or multi-layered fabric assembly along border parts (seams, darts, etc.);
- (*manufacturing*) border/interior finishings (stitches, etc.);
- (*manufacturing*) insertion of small functional/aesthetic rigid or soft components (e.g., buttons, hooks, zips, paddings, etc.);
- (*manufacturing*) mechanical or chemical post-treatment of garment textiles (pleats, ironing, starching, etc.);
- (*design/manufacturing*) 3D configuration/placement of the garment over supports or external objects (e.g., mannequins);
- (*final evaluation*) analysis of the final garment shape and behaviour in the 3D physical space.

Our idea is to propose a computer-assisted methodology in which the garment definition process is analysed throughout the above steps. In the present work we are concerned with design and evaluation phases, but do not consider manufacturing issues. Our most significant contribution is the definition of a complete physics-based model built coherently with the above design steps. The main intent is performing an accurate simulation of garment's behaviour for virtual prototyping tasks (corresponding to the *final evaluation* item). In the next Sections we present the system that has developed according to these aims.

4. THE PARTICLE-BASED MODEL OF GARMENTS

As a physical model, a discrete "particle-based" approach is chosen (Frugoli et Al. 2000), in which cloth is approximated with sets of particles located at the interior and at the border of fabric panels, according to a specific resolution, and connected with each other according to structured grids, similarly to Breen's (House and Breen 2000) and Provot's (Provot 1997) models. The internal material behaviour is described by inter-particle force links computed as linear or torsional springs to describe traction, compression, bending and shear effects. The spring constants are estimated from experimental values derived by the Kawabata Evaluation System (Kawabata 1980), measuring the mechanical behaviour of fabric hands depending on material (Figure 2).

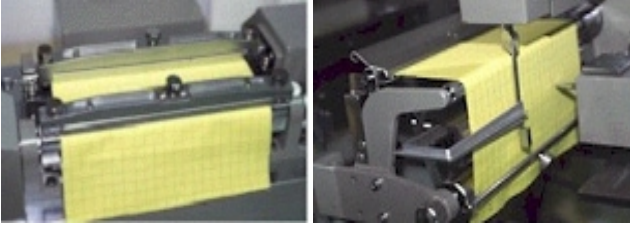


Figure 2. KES-F measurements of a fabric sample.

To obtain the complete particle-based model of the whole garment, the processes of sewing, layer overlapping and button insertion need to be applied to the original particle grids associated to each single layer. In the sewing process between two panels, border particles are properly merged after displacement, removal and possible addition of particles over the edges to be sewn. Layer overlapping causes a change in the values of particle masses and internal force values within the layer, while insertion of buttons is emulated with mass increment and addition of hard spring forces for particles located in proximity of buttons.

Garment's dynamical behaviour can then be predicted by analysing it as a system of particles subjected to Newton's laws of motions, as described in the algorithm in Section 5.3.

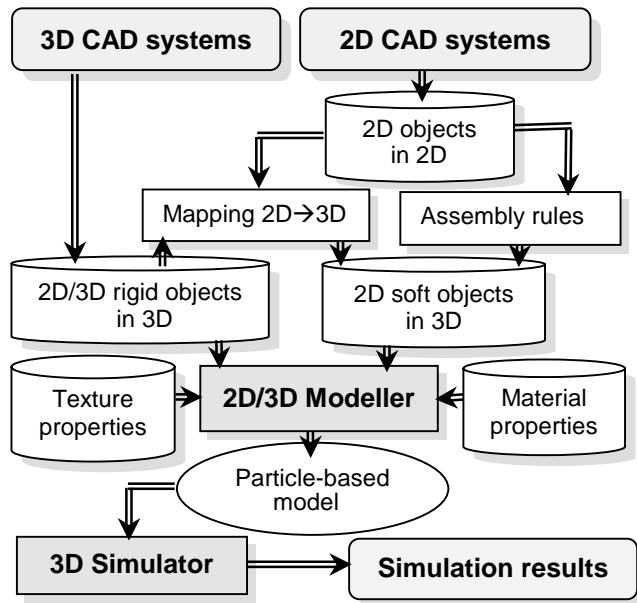


Figure 3. Architecture of the system.

5. THE PROPOSED SYSTEM

5.1 The general architecture

Our system, specifically targeted for virtually simulating fabrics and garments, is composed of several modules for data conversion, geometric modelling, physics-based modelling and dynamic simulation. Besides, it is provided with a Graphical User Interface. The general high-level architecture is shown in Figure 3. Grouping together the several modules by scope, two fundamental procedures are singled out:

- *2D/3D Modeller*, generating the physics-based model associated to 2D textile objects in 3D scenes;

- *3D Simulator*, returning the dynamic simulation of the scene, with frames computed at time steps t_0, t_1, \dots, t_n .

5.2 2D/3D Modeller

The 2D/3D Modeller associates a physical model to a 3D scene containing textile configurations and rigid objects, by considering a particle-based discretization for deformable components (textiles, garments, etc.) and a corresponding point-based discretization for rigid objects (mannequins, supports, frames, etc.). The 3D configuration of garments and other textile objects to be discretized by particles has been obtained during a previous geometric modelling phase, from panel sewing, layer overlapping, dart/button insertion, and mapping of 2D panels onto 3D mannequins or other rigid supports. To give a cloth physical representation, it is necessary to specify the material composition of fabrics. The input data structure of the 2D/3D Modeller can be thus summarized as follows-

- material properties (e.g., KES data);
- rigid bodies (e.g., mannequins, frames);
- geometry of 2D panels;
- rules for 2D→3D mapping;
- model construction constants (e.g. textile operations: seams, darts, fabric layers, buttons);
- (if any) dynamic constraints (kinematics of rigid parts).

As regards the data returned in output by the Modeller, they can be summarized as:

- physics-based information about textile objects (3D particle positions, particle masses, associated grid, internal forces, external forces, geometrical and kinematic constraints);
- physics-based information about rigid parts (3D point positions, associated grid, geometrical/kinematic constraints).

5.3 3D Simulator

The output data of the 2D/3D Modeller are used as inputs for the 3D Simulator, which computes the dynamic behaviour of the physical model as a particle system subjected to constrained Newtonian dynamics. The external environment affects the behaviour of the particle model by means of exerted external forces, imposed geometrical or kinematic constraints and collisions. The output data of the Simulator are physical data at discrete time steps t_0, t_1, \dots, t_n describing:

- configurations $\{\mathbf{r}_i(t_k)\}$ for each particle $i=1,2,\dots,N$, for $k=1,2,\dots,n$;
- velocities $\{\mathbf{v}_i(t_k)\}$ for each $i=1,2,\dots,N$, for $k=1,2,\dots,n$;
- other information: forces, energies, etc.

The main simulation algorithm can be summarized as follows.

Known initial configuration $\{\mathbf{r}_i(t_0), \mathbf{v}_i(t_0)\}$, $i=1, 2, \dots, N$.

For step $k=1, 2, \dots, n$:

1. It computes the active forces $\mathbf{F}_i^{(A)}(t_k)$ (from internal and external contributions) on each particle i ;
2. computes Lagrange multipliers $\{\lambda_j\}$, for all bilateral constraints (SL solver);

- estimates the reactive forces $\mathbf{F}_i^{(R)}(t_k)$ from Lagrange multipliers for all particles i (equal to zero if no constraints involve i);
- computes the new velocities $\mathbf{v}_i(t_k)$ from total forces $\mathbf{F}_i^{(A)}(t_k) + \mathbf{F}_i^{(R)}(t_k)$, for all i , by numerical integration of Newton's law (ODE solver, 1st part);
- detects the colliding particles;
- computes $\Delta \mathbf{v}_i$ from collision response algorithm (LS solver), with $\Delta \mathbf{v}_i = 0$ for uncolliding particles;
- updates velocities $\mathbf{v}_i(t_k)$ with $\mathbf{v}_i(t_k) + \Delta \mathbf{v}_i$;
- computes the new particle positions $\mathbf{r}_i(t_k)$ from updated velocities (ODE solver, 2nd part).

Details about collision detection, analytical methods for constraint management and collision response, as well as numerical issues related to ODE and SL solvers, are detailed in (Frugoli et Al. 2000; Galimberti 2002).

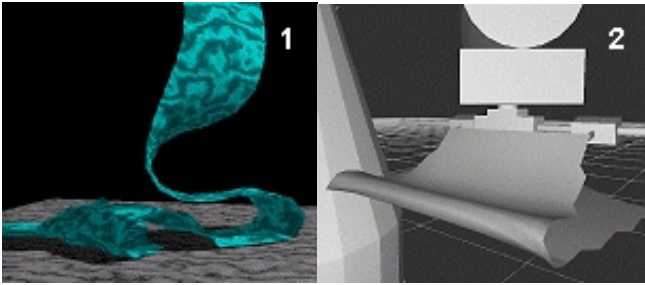


Figure 4. Simulation of fabrics on obstacles: (1) Ribbon falling on the floor. (2) Fabric layer handled by a robot.

6. APPLICATIONS

The system has been first evaluated successfully on simple test cases, such as the simulation of tablecloths over a table, a sphere, or other rigid supports, ribbons on the floor, fabrics handled by robots, etc. (Figure 4).

Successively, as our idea is to validate the system within a CAD framework for designing real cloth, we have analysed garment prototypes presenting design complexities such as seams, complex borders, multi-layered fabrics, different materials, etc. Modelling approaches and results for female and male garment production are shown in the following.

6.1 Female Apparel

The system has been first experimented in the framework of the Brite-Euram Project n° 3542, named MASCOT (Intelligent 3D Design and Simulation System for the Clothing Industry), aiming at developing an integrated 2D/3D design and simulation system for the apparel production. Activity has been carried out for the female sector (Bonte et Al. 2000).

Modelling and simulation algorithms developed by KAEMaRT group have been accompanied here by a CAD interface for 3D geometrical modelling, provided by the University of Valenciennes. In the modelling phase, initial 3D dress models are directly chosen from a library of template cloth shapes (skirt, blouse, etc.), then modified by NURBS surface modelling tools, acting on characteristic lines. Seams, pences and other textile characterizations are defined on the 3D model, from which the 2D panels are

extracted for successive particle-based modelling using material data in KES format. 2D panel's particle grids are then sewn and located properly on the mannequin for the final simulation. In Figure 5, as examples, we display some modelling sequences to create the surface models of female garments before the physics-based modelling phase. Figure 6 presents the simulated results of a dress, a skirt and a linen tunic. Details about the modelling and simulation phases can be read in (Galimberti 2002).



Figure 5. Free form modelling of female garments.

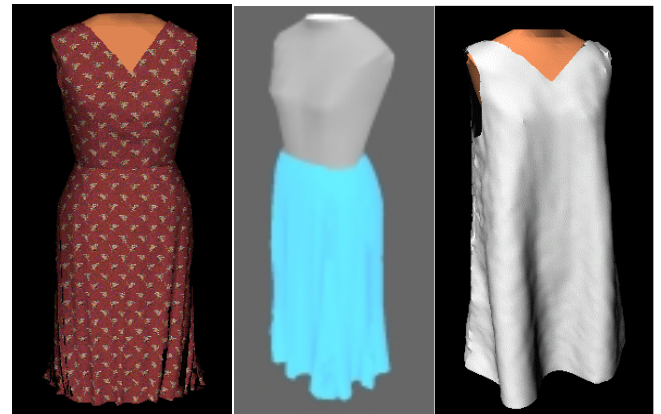


Figure 6. Simulated female garments.

6.2 Male Apparel

A second activity has been carried out within a national project, named Consorzio TA2000 per l'Industria della Moda (Textile Apparel Consortium 2000 for the Fashion Industry), in collaboration with several Italian clothing companies (Cugini and Rizzi 2002). One of the main tasks in this project is developing software tools and methods for virtual cloth prototyping in male garment production.

Male fashion is characterized by more standardized shapes, with respect to female's. At each season, models of jackets, trousers, shirts are proposed, presenting just few modifications in small aesthetic shape details, and material/color combination. For this reason, the initial CAD models have been defined from digitalization of existing apparel articles. Jackets have been considered as test cases. Successive smoothing of the digitalized jacket's point clouds

leads to optimized 3D models. A modification interface, based on Maya Deformer tools (from Alias|WaveFront), has been implemented ad hoc to allow interactive modification of functional parts of jackets' models (sleeves, shoulders, breast, waist, etc.), with free form deformation driven by characteristic lines.

Export modules automatically generate files of the resulting model in proper format, used as input for 2D panel extraction. Generation of 2D panels is done by means of an owner textile CAD software used at Corneliani clothing industry. Panel's particle-based grids are obtained through system's Modeller, then assembled together on virtual mannequins by using previously stored 2D→3D mapping rules. Once the particle-based model has been created in 3D, physics-based simulation can start.

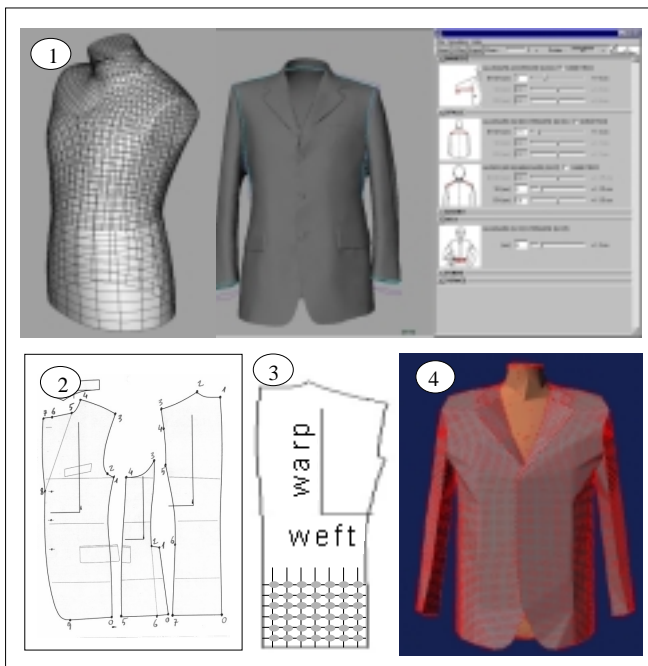


Figure 7. Virtual design of male garments. Main modelling steps: (1) 3D modeling module. (2) Extraction of 2D patterns. (3) Particle-based model of each panel. (4) Final assembled particle-based model ready for 3D simulation.

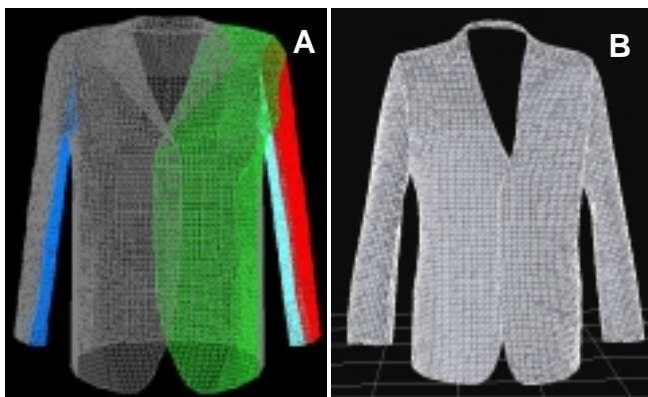


Figure 8. Jacket's particle-based models: (A) Configuration before the simulation. (B) Final result after the simulation. Different colours in A denote different panels sewn together

Figure 7 displays the main design steps. Figure 8 displays the initial wireframe configuration of jacket's particle-based model before the simulation, and the final configuration at equilibrium after the simulation. Details of the simulated jacket are shown in Figure 9: pleats and side vents (A-C), padded shoulders (A-B), lapel and collar (B). As expected, differently from the considered female models, finer particle meshes have been used for the examples of jackets to reach a good accuracy. In fact, jacket models have more complex-shaped configurations with a large number of panels, multi-layered fabrics (e.g., linings and stuffings), and multiple sewings. Details about the jacket model and simulation results can be read again in (Galimberti 2002).

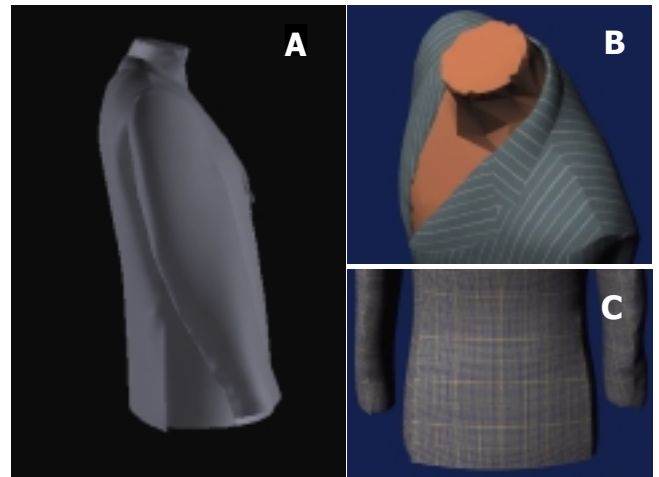


Figure 9. Some views of a simulated jacket.

7. CONCLUSIONS AND FUTURE WORK

A system and a CAD methodology were presented for virtual garment design and simulation. As applications, several women's and men's garments were modelled and simulated in 3D virtual scenes. Different models of dresses, skirts, jackets were simulated, to include several levels of design complexities, e.g., different material composition, multi-layered fabrics, presence of seams, buttons, etc. The underlying physical model was based on a discrete cloth representation through structured particle grids subjected to constrained Newtonian dynamics.

Work is currently in progress for expanding the geometric modelling capabilities of the system and specializing them to the textile context, e.g. computer emulation of tight/loose seams and manufacturing effects (e.g., pressing, pleat generation, starching, etc.). Efforts are currently under way to improve simulation performances and reduce the computation time, by considering optimization techniques for collision detection and efficient implicit and semi-implicit ODE solvers for the numerical solution of Newtonian particle systems associated to garment configurations, e.g. (Etmuss 2001).

ACKNOWLEDGEMENTS

This work has been carried out in the framework of the Brite-Euram Project MASCOT, funded by the European commission, and the Italian Project TA2000, funded by MIUR.

REFERENCES

- Bonte, T., Galimberti, A. and Rizzi, C. 2000. "A 3D graphic environment to design garments", *From Geometric Modeling to Shape Modeling*, U. Cugini & M. Wozny Eds., Kluwer Academic Publishers, 137-150.
- Cugini, U., Bordegoni, M., Rizzi, C., De Angelis, F. and Prati, M. 1999. "Modelling and haptic interaction with non rigid materials", *Proc. Eurographics '99*, State of the Art Reports, 1-20.
- Cugini, U. and Rizzi, C. 2002. "3D Design and Simulation of men garments", *Proc. Workshop WSCG '02*.
- Etzmuss, O., Keckeisen, M., Kimmerle, S., Mezger, J., Hauth, M. and Wacker, M. 2001. "A Cloth Modelling System for Animated Characters", *Proc. Graphiktag '01*, 1-10.
- Frugoli, G., Galimberti, A., Rizzi, C. and Bordegoni, M. 2000. "Discrete element approach for non-rigid material modelling", *Robot Manipulation of Deformable Objects*, D. Henrich & H. Worn Eds., Advanced Manufacturing Series, Springer, 29-42.
- Galimberti, A. 2001. "Sviluppo di un sistema di simulazione per la progettazione di capi di abbigliamento", *Ph.D. Thesis*, Dipartimento di Ingegneria Industriale, Università di Parma, Italy.
- Hinds, B.K. and McCartney, J. 1990. "Interactive garment design", *Visual Computer*, Vol. 6, 53-61.
- House, D.H. and Breen, D.E. Eds. 2000. "Cloth modelling and Animation", A K Peters Ltd., Natick, MA, US.
- IEEE Computer Graphics and Applications Eds. 1996. Special issue on "Computer graphics in Textiles and Apparel", Vol. 16, N. 5.
- Kawabata, S. 1980. "The Standardization and Analysis of Hand Evaluation", The Textile Machinery Society of Japan, Osaka.
- LAMIH. 1999. "Architecture of 3D modeller", Technical Report, *MASCOT Brite-Euram Project N. 3542*.
- Leung, K.Y.C, Taylor, G., Yuen, M.M.F. and Kung A. 1997. "Development of Three Dimensional Computer-Aided-Design System for the Garment Industry", *Proceedings of the 4th Asian Textile Conference*, Taipei, 1025-1029.
- Okabe, H., Imaoka, H., Tomira, T. and Niwaya, H. 1992. "Three Dimensional Apparel CAD System", *Computer Graphics*, Vol. 26, N. 2, 105-110.
- Provot, X. 1997. "Collision and self-collision handling in cloth model dedicated to design garments", *Proc. Graphics Interface '97*, 177-189.
- Vassilev, T., Spanlang, B. and Chrysanthou, Y. 2001. "Fast Cloth Animation on Walking Avatars", *Proc. Eurographics '01*, 260-267.
- Volino, P. and Magnenat Thalmann, N., Eds. 2000. "Virtual Clothing. Theory and Practise", Springer-Verlag, Heidelberg, Germany.

BIOGRAPHY

MARZIA FONTANA is a Ph.D. candidate at the Department of Industrial Engineering, University of Parma, Italy. She received her M.S. Degree in Mathematics at the same university and was awarded with grants for research programs at the Institute of Applied Mathematics, CNR, Genoa, Italy, and at Chalmers University of Technology, Goteborg, Sweden. She is author and co-author of several publications in the area of mathematical physics, finite element methods, surface modelling and physics-based modelling.

CATERINA RIZZI is full professor at the Faculty of Engineering, University of Bergamo, Italy. Previously, she was professor at the Universities of Naples and Parma, Italy. She participated at several national and international projects in the area of modelling, simulation and CAD applications. She is author and co-author of more than 80 publications in the fields of CAD, physics-based modelling and simulation, knowledge-based modelling, virtual prototyping, cloth modelling.

UMBERTO CUGINI is full professor at the Department of Mechanics, Polytechnic of Milan, Italy. Previously, he was researcher at CNR, Milan and, successively, professor at the University of Cagliari and Parma, Italy. He participated at several national and international projects in the area of modelling and simulation, and directed several national projects and consortia in the area of design and robotics. He is author and co-author of more than 150 publications in the fields of CAD, industrial design, modelling and simulation, knowledge-based modelling, virtual prototyping and robotics.

REAL-TIME VISUALIZATION OF WOVEN TEXTILES

Neeharika Adabala
MIRALab - University of Geneva,
24 Rue General Dufour,
CH1211 Geneva 4,
Switzerland
adabala@miralab.unige.ch

Nadia Magnenat-Thalmann
MIRALab - University of Geneva,
24 Rue General Dufour,
CH1211 Geneva 4,
Switzerland
thalmann@miralab.unige.ch

Guangzheng Fei
Animation School,
Beijing Broadcasting Institute,
China
gzfei@bbi.edu.cn

KEYWORDS

Weave Patterns, Illumination Model, Multi-Texturing.

ABSTRACT

This paper presents a technique for visualizing woven textiles in real time, while optimizing the realistic appearance. The proposed approach supports rendering of complex weave patterns by adopting Weaving Information File (WIF), a standard from textile Computer Aided Design (CAD) for representing the grammar of weaving. We develop a realistic rendering scheme by combining the grammar representation obtained from the WIF with a procedural thread texture, a suitable textile Bi-directional Reflectance Distribution Function (BRDF) and horizon maps. We employ the multi-texturing approach to meet the real time constraint. Thus our approach to visualizing woven textiles begins from weaving grammar specifications and converts them into textures that can be used for visualizing clothes. We demonstrate the versatility of the proposed approach with examples.

INTRODUCTION

There are two main aspects to the problem of modelling and simulating textiles:

- **Modelling the dynamics of textiles:** Textiles are flexible and exhibit dynamic behavior when interacted with. This is an important characteristic of textiles that has been studied extensively and a large amount of research has been done in the area of modelling dynamics of clothing in the context of textile research (Kawabata 1975; Clapp and Peng 1990; Collier et al. 1991; Gan et al 1995; Kang and Yu 1995) and computer graphics (Volino and Magnenat-Thalmann 1995; Baraff and Witkin 1998). Most of these techniques are non real-time, however recent work (Cordier and Magnenat-Thalmann 2002) has made it feasible to simulate the clothing dynamics in real time.
- **Visualizing textiles:** Textiles exhibit micro and macro-geometry (we refer to milli-scale geometry as macro-geometry in this paper). The light interaction with the interwoven threads has to be captured in order to create a realistic appearance.

This paper focuses on the second aspect namely visualizing textiles. Our goal is to create a real time technique that enables rendering of complex weave patterns. The ability to

visualize the appearance of the textile realistically is important for the completeness of digital representation of textiles. Such digital virtual textiles have application in fashion industry where computer aided prototyping of clothing is gaining importance, in e-commerce where an authentic representation of real world articles is needed, in clothing for virtual character, etc. We have chosen to meet real-time constraints in order to be able to integrate our approach with clothes designing software, which requires the designer to perceive changes immediately.

There are two main types of textiles namely, the knit and woven textiles. Impressive results have been reported in recent research on visualization of knitwear (Zhong et al 2000; Daubert et al. 2001; Gröller et al 1995; Meissner and Eberhardt 1998; Xu et al 2001). Techniques to represent complex knit patterns have been developed (Meissner and Eberhardt 1998, Zhong et al 2000). Relatively less work has been done in the area of woven textiles, especially in the context of representing complex weave patterns. The main issues that have to be addressed when developing a technique for visualizing textiles are:

- Representation of interwoven threads
- Modelling of light interaction with the threads/ textile.

We briefly describe some of the previous work in the light of these issues. Westin et al. (Westin et al 1992) consider a weave pattern and obtain realistic rendering of textile. In this work they consider a simple alternate weave pattern known as the linen binding and model the illumination at micro-scale and extend it to the milli-scale by performing an integral over the surface. Yasuda et al. (Yasuda et al. 1992) describe a shading model for textiles that emphasizes on the interaction of light with individual fibers that constitute the textile. Gröller et al. (Gröller et al 1996) use a technique based on three-dimensional textures to model textiles. More recently, Daubert et al. (Daubert et al. 2001) have presented an efficient technique for modelling and rendering clothes, their approach is applicable especially for coarsely woven fabric and knitted fabric.

Apart from the work that directly addresses the problems of rendering clothes there is a class of work based on light interaction with surfaces that exhibit micro-geometry that can be applied to address problems in rendering of clothes. These include the work by Ashikhmin et al. (Ashikhmin et al. 2000) that develops a micro-facet based technique for modelling light interaction with surfaces and applies it

effectively to simulate the appearance of velvet and satin. Also the techniques of illuminating micro-geometry described in Heidrich et al. (Heidrich et al. 2000) and Sloan et al. (Sloan and Cohen 2000) find direct application to visualization of textiles as they exhibit micro-geometric details.

All the existing techniques are very powerful and solve the problem of visualizing clothes effectively, but the existing techniques do not address the problem of being able to capture a wide variety of weave patterns. Some variations other than the most common linen binding have been considered in (Ashikhmin et al. 2000) and (Daubert et al. 2001) where patterns similar to satin binding are considered. However the more complex ones have not been addressed.

The objective of our technique is to provide clothes and graphics designers with the capability of generating textures with various designs of weave patterns to texture the clothes designed by them without having to use scans/photographs of sample textiles. Approaches using scans/photographs suffer from the problem of presence of features resulting from the illumination under which the images were generated. Thus there is a need to provide textures with the ability to exhibit the change in illumination behavior, without the illumination being incorporated into the color of the material in a fixed way. Also, we would like to zoom in onto the textile and be able to perceive detail. In order to address these issues we break up the problem of visualizing textiles into:

- Representing the weave pattern
- Modelling the microstructure of the threads of the textile.
- Modelling the light interaction with the textile:
 - Reflection behavior

- Shadowing behavior between the threads in the weave pattern.

Our approach addresses the above issues as follows: We use a standard in the textile industry known as the Weaving Information File (WIF) to obtain the complex weaving patterns and represent patterns suitably for generating the color texture. The microstructure of the threads is incorporated into a procedural texture and it is used along with the WIF information to generate a color texture. The WIF information is also used to define a suitable Bi-directional Reflectance Distribution Function (BRDF) that corresponds to the reflection behavior of the textile. The shadowing that occurs between the threads when they are woven together is captured in the horizon maps that are generated with information from the WIF. We are thus able to start from the grammar representation for weaving a textile and generate a visualization of the woven textiles. One of the important aspect of the work is that we not only capture the appearance based on the weave grammar but we also capture the way light interacts with the material.

The organization of the paper is as follows: the next section an overview of our algorithm is given. The details of the four main parts of our algorithm are then described. We discuss the results by presenting example images that demonstrate the capabilities of our algorithm. Conclusions and suggestions for future work are given in the final section.

OVERVIEW OF THE ALGORITHM

The outline of the algorithm is presented in the Figure 1.

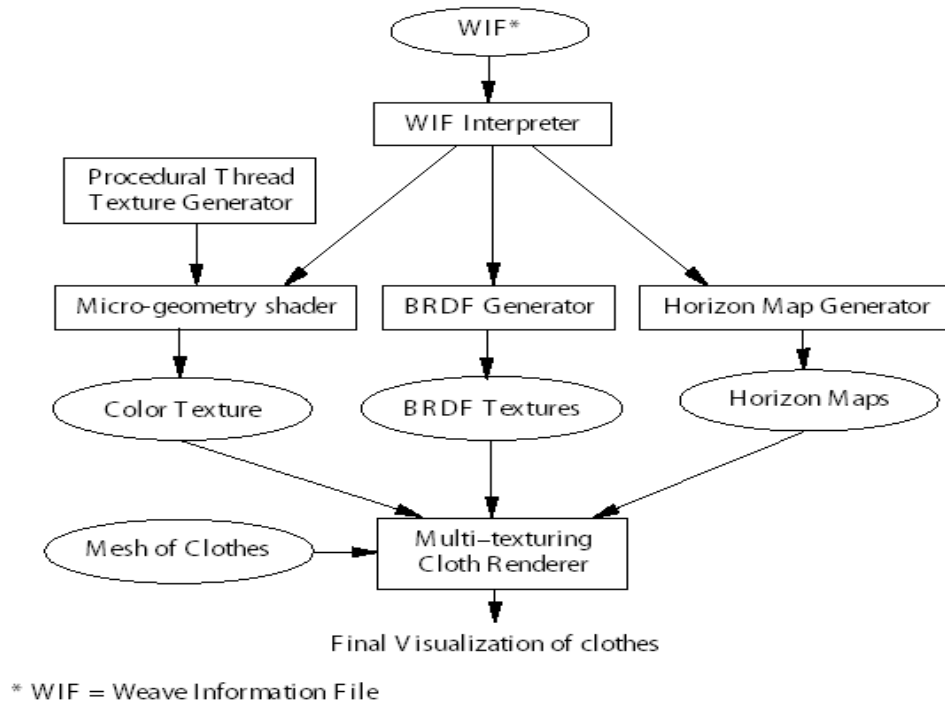


Figure 1: Outline Of Algorithm

The algorithm uses the information in WIF that describes the weave pattern. It is interpreted by the WIF interpreter and made available to the modules namely: the micro-geometry shader, the BRDF generator and the horizon map generator. The micro-geometry shader makes use of a procedural thread texture generator to create a color texture based on the WIF. The procedural thread texture generator is responsible for creating the shading that results from the twisting of the fibers that are spun into the threads.

The BRDF generator takes as input the characteristic of the textile obtained from the weave pattern described in the WIF. It generates a BRDF to correspond to these characteristics. The WIF does not contain any properties that describe the illumination characteristics of the thread, therefore WIF alone does not give the complete information required for defining an illumination model. However, the pattern of weaving contributes significantly to the way the light interacts with a fabric. This is very well exemplified by the fact that the satin weave results in textiles with glossy appearance.

The WIF contains the weave grammar and therefore from this information it is possible to learn which facets of the threads are longer and hence tend to be higher on the surface of the textile. This in turn dictates how such facets of textile cast shadows on the neighboring threads of the textile. The horizon map generator module uses the WIF information to define these shadows that are essential to convey the feeling of depth in the weave pattern. The real time constraint prompted us to use the multi-texturing approach to realize the solution to our algorithm. Each of the above modules results in a texture and the final rendering of the textile is done by compositing the images suitably.

DETAILS OF THE ALGORITHM

The four main modules of our approach namely the WIF interpreter, the micro-geometry shader, the BRDF generator and the horizon map generator are described in this section.

WIF Interpreter

Woven fabrics exhibit well-defined structures; it should be possible to use a procedural or grammar-based technique to represent the weave pattern. However, the wide variety of the weave patterns limits the applicability of such techniques. Fortunately, in CAD of textile there is a well-established technique for representing the weave pattern as WIF format (Nielsen et al. 1997). It is a specification that provides the information required for weaving a fabric in textile looms. The WIF includes information from which the weave pattern can be derived. The pattern indicates how warp and weft threads are interwoven

Since the WIF format was designed for manufacturing purpose rather than for visualization, it is not directly applicable to computer graphics. The WIF contains the threading information that defines which warp thread goes through the heddle in which shaft. It also contains a liftplan that represents the combination of shafts raised for creation of each weft. The weave pattern can be obtained by

combining the threading and the liftplan information. We parse the WIF format and derive the weave pattern from it. The weave pattern is represented as a two dimensional matrix, where the rows and the columns can be thought of to index the weft and warp threads respectively. Each entry in the matrix indicates the visibility of the weft or warp thread at that point. The number of weft and warp threads present in the weave pattern determines the dimension of the matrix.

The WIF format also contains color information for each thread that can be directly combined with the pattern matrix to generate the color scheme for the weave pattern. Since the weave pattern matrix indicates which thread is seen at each point on one face of the textile, the texture for the other side of the textile is easily obtained by complementing the matrix. Figure 2 shows a color scheme of a complex weave pattern generated from a WIF format.

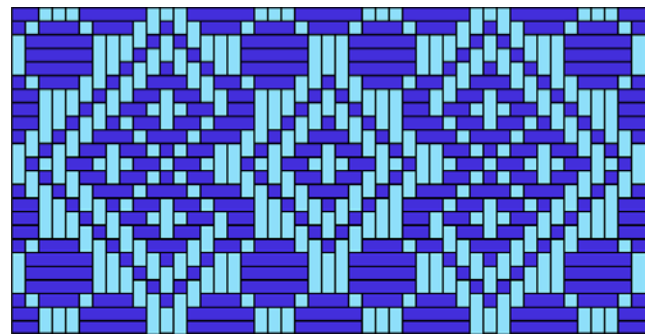


Figure 2: Example Color Scheme Of A Weave Pattern

Micro-geometry Shader

The woven fabrics are made up of interwoven twisted fibers. We have observed that when one examines woven textiles at the usual distances of viewing the twisted nature of the thread facets is often visible. The visibility is caused by the presence of dark shaded lines that follow the twist of the fibers of the thread. In some cases these lines are seen prominently while in other cases they are less prominent. It is possible to discern whether the threads are tightly or loosely twisted from these shaded lines. We also observed that the shading present on the thread facet tends to remain approximately the same under various lighting conditions. We may attribute this to presence of deep fine groove between the twists of the fiber into which the light never reaches. This feature is unlike the other illumination aspects of the macro-geometry of textiles in the form of shadowing between the threads that show variation with position of light. Also unlike wool where the fibers occupy a significant volume, the threads that are woven into a textile are finer and are limited to a near two-dimensional surface.

The above observations lead us to separate the micro and macro-geometry details present in the woven textiles. We exploit the near two-dimensional nature of the textile surface by modelling the facet of thread visible on the surface of a textile as a two dimensional procedural texture. We design the procedural texture such that it has parameters to capture the tightness of the twist and thickness of the thread. Figure 3 shows examples of the thread shading textures that are generated procedurally by our technique.

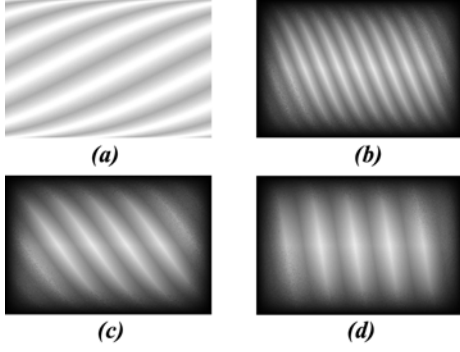


Figure 3: Output Of Procedural Texture (a) Very Loosely Twist Of Thread Without Shading. (b) More Tightly Twisted Thread, Noise Is Added To Simulate The Presence Of Fibers About The Thread. (c) Thicker Fibers Twisted Into Thread. (d) Tightly Twisted Thread.

These thread textures can be used along with the weave pattern to generate a color texture of a weave pattern that has the appearance of being woven from twisted fibers. An example of such a texture is shown in Figure 4

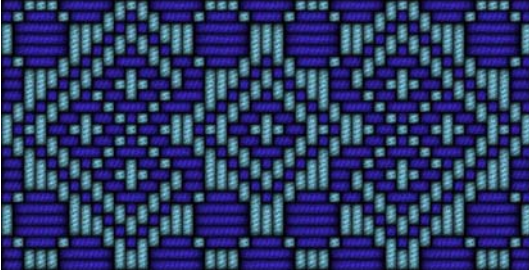


Figure 4: Color Texture Generated For The Color Scheme

BRDF Generator

The image that is generated by combining the micro-geometry shader and the weave pattern does not include an illumination model. It only contains the colors of the threads and the pattern of weaving. However, the same weave pattern can have a significantly different appearance based on the material of the threads that constitute the textiles. Also, the same threads can result in a different texture of the textile depending on the weave pattern. This has already been pointed out before when it was mentioned that use of the satin weave pattern results in a more glossy texture than linen weave pattern for the same quality of thread.

In our technique the BRDF generator makes use of the WIF information to define a BRDF. Various approaches to represent the BRDF can be adopted. We choose the micro-facet based BRDF modelling (Ashikhmin et al. 2000) because it is flexible and can be used to model complex reflection behavior. In this approach design of suitable probability distribution functions of micro-facet enables modelling of various types of textures.

We have define a generalization of the formulation used in for satin in [1], to represent the probability distribution. The probability distribution of the micro-facets that we use is:

$$p(\vec{h}) = f_{warp} * p_{warp}(\vec{h}) + f_{weft} * p_{weft}(\vec{h})$$

where $p(\vec{h})$ represents the probability distribution of the normals of the micro-facet, f_{warp} and f_{weft} are respectively the fractions of the surface occupied by the warp and weft threads. The probability distributions of facets on individual warp and weft threads are given by the $p_{warp}(\vec{h})$ and $p_{weft}(\vec{h})$ respectively. The \vec{h} represents the normalized half vector between the vector to the light and the vector to the viewer. The parameters f_{warp} and f_{weft} are computed from the WIF. However no information is present in the WIF to enable us to define the probability distributions of the micro-facets on individual threads in the warp and weft directions, namely the functions for $p_{warp}(\vec{h})$ and $p_{weft}(\vec{h})$. We use a *cylindrical* Gaussian with $\sigma_y = \infty$ similar to the one described in [1] for this purpose. The width of the thread is used to choose the σ_x of the cylindrical Gaussian. The details for evaluating the BRDF from a distribution of microfacets are found in (Ashikhmin et al. 2000).

The textiles can appear more biased to the color of the warps or wefts depending on the angle of viewing. This view dependence of the appearance of the cloth is incorporated into the BRDF when the colors that are woven together are very contrasting with the help of the Fresnel's co-efficient. The co-efficient is computed as a weighted sum as follows: When the angle of viewing is close to perpendicular the warp color is given a higher weight. When the angle of viewing is at grazing angles the weft color is given a higher weight. The cosine of the angle between viewing direction and the normal to the surface is used to compute the weight. This is a heuristic approach to incorporate the dependence of appearance on the viewing angle.

We achieve the real-time rendering of the BRDF by using the cube map approach (Kautz and McCool 1999). The BRDF we use is defined for the whole textile rather than for individual thread segments therefore some details like shadowing of threads on each other are not captured by it. These details are captured in the horizon maps that are described in the next section.

Horizon Map Generator

Horizon maps store shadows cast by small surface perturbations on itself. In the case of fabrics in outdoor day light scenes this feature is relatively less important as there is a large amount of light incident on the fabric from all directions resulting in the absence of shadowing among threads. However, in the case of artificial lighting in indoor scenes we have observed that the fabrics tend to look significantly different under lighting and this is due to the shadows cast by the threads on each other.

We observe that the height of a facet of thread above the textile surface is dependent on the length of the facet on

which it occurs. However, the height is limited to a maximum level dictated by the tightness of the weave. This height in turn defines the shadow that it can cast on the neighboring thread facets. The WIF information in the form of the weave pattern matrix is used to compute the length of the facet at each location within the weave. This information is then translated into a height field that is further used to compute the horizon maps. We discretize the directions of the light and generate a shadow map for each of the directions. This approach is less accurate than the techniques proposed in [7] and [11], however we found that it gives reasonable results for our real time constraints.

RESULTS AND DISCUSSION

This section presents example images that have been generated using our technique. We implement our algorithm on a PC with a pentium 4 processor (2 GHz), with a nVIDIA graphics card. The clothes that have been rendered in this

section were modelled with MIRACloth, the in house software of MIRALab for designing clothes. All the models in the figures consist of approximately 2000 triangles each. We perform the computation of the color texture, BRDF texture and the horizon maps as a preprocessing step. The computation time for the color texture is less than one second and each of the shadow maps takes about half a second. The BRDF computation along with the separation for cube maps requires around 140 seconds. The rendering of the images from these pre-computed textures using multi-texturing is achieved in real-time.

We present example images of jackets created from various weave patterns in Figure 5. Use of the WIF based method enables easy generation of many complex and interesting weave patterns. This result is especially relevant to show that our technique combines the benefits of developments in both computer graphics and textile industry.

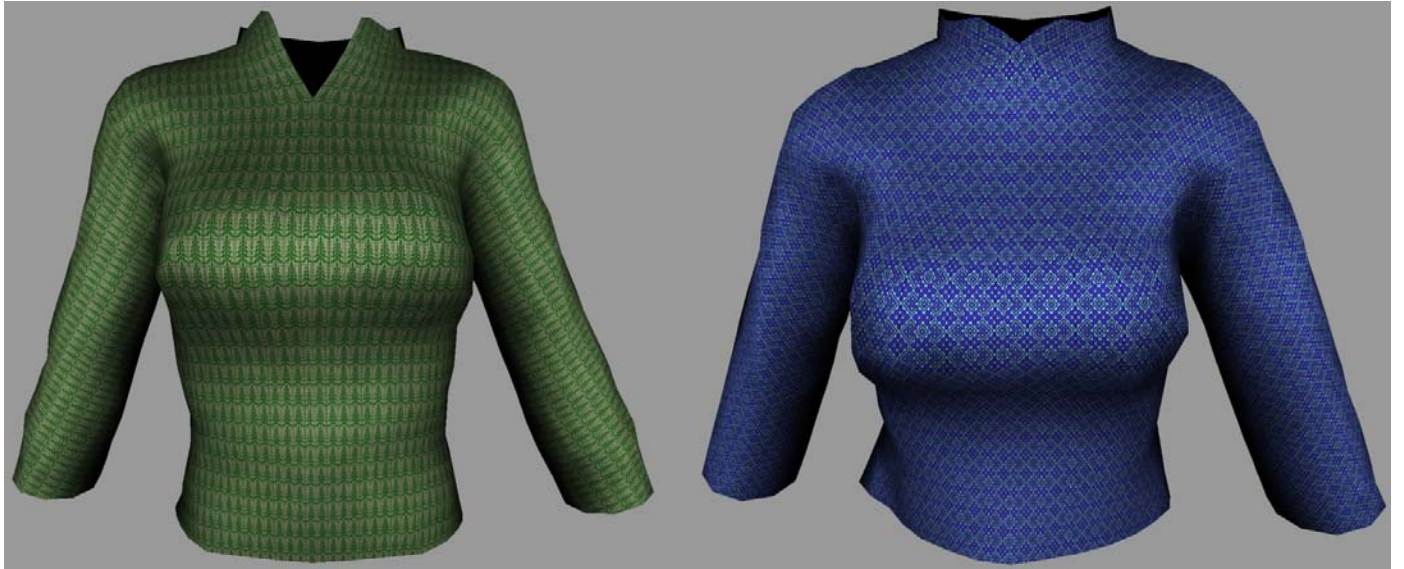


Figure 5: versatility of the approach for generation of various weaves. The jacket on the right is textured by the weave pattern presented in figure 4.

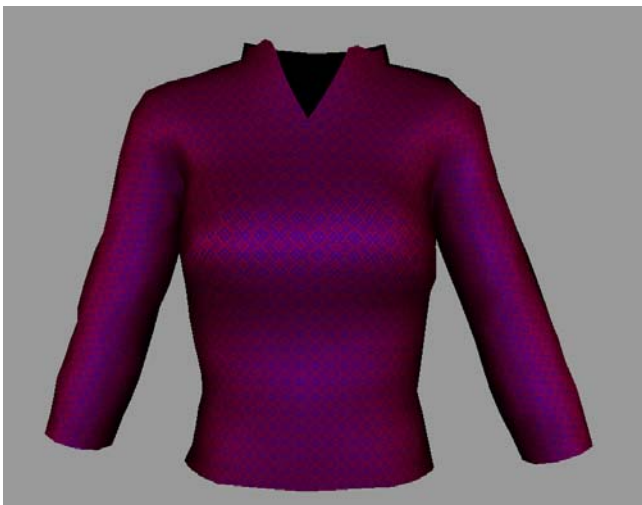


Figure 6: directional dependence of the appearance of textile when contrasting colored threads are woven together.

The figure 6 shows the power of our illumination model. We are able to capture the variation in appearance of a cloth woven with contrasting colored threads. This is possible because of our BRDF function that models the detailed light interaction. This result is not possible with only texture-based approaches that are widely used. The ability to visualize such textiles realistically gives a designer a clearer perspective of the end product of their design.

We demonstrate the ability of our technique to support viewing at various levels of detail in the Figure 7. It can be seen that our technique results in a very realistic appearance of the textile even at close distances. The presence of shadows enhances the feeling of thickness of the textile.

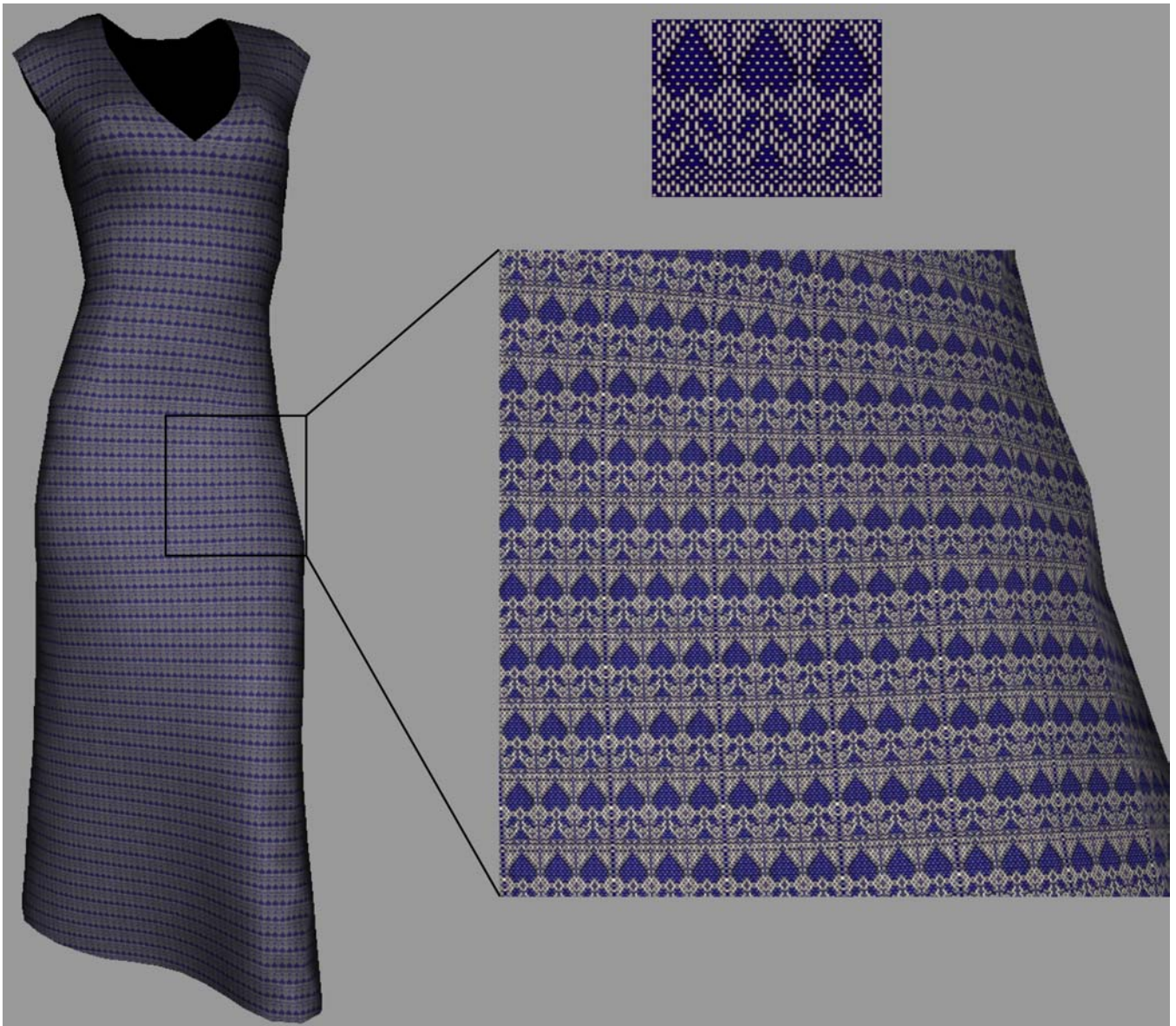


Figure 7: illustration of ability to zoom in on detail. The image on the top right gives the weave pattern used to texture the dress.

CONCLUSIONS AND FUTURE WORK

This paper has proposed a technique for generating realistic visualization of woven textiles in real-time. The main features of the work are:

- **Texture of fabric generated from grammar:** Such textures are typically generated as sample bitmaps with no accompanying illumination model. In our case we have developed a more complete representation of the textile that also models the light interaction. As it works in real-time it can be used for visualizing textiles in clothing design.
- **Compatible with textile CAD:** Use of the WIF format that has been a standard in the textile industry from 1996, makes our work compatible with work carried out in the textile industry and thus helps to bridge gaps by sharing knowledge between the computer graphics technologies and textile technologies.
- **Micro and Macro detail separation:** In order to meet the real time constraint and also not lose the details in

modelling the cloth, we separated details of the cloth visualization into micro and macro details. The micro details were incorporated into the thread textures that were generated procedurally while the macro details were incorporated into the BRDF and horizon maps.

- **Weave pattern based BRDF:** The ability to represent the weave as a grammar enabled us to define the BRDF based on the pattern. Relatively less importance has been given to this aspect of visualizing textiles in computer aided design tools of the textile industry. However this is of prime importance for realism of the visualization and here again there is a bridging of gaps between the two areas.
- **Works in real time:** The multi-texturing approach was exploited to develop an algorithm that works in real-time.

While the algorithm described in the paper gives realistic results, the BRDF technique presented in the paper uses an estimation over the whole weave pattern. However, it may be possible to investigate a more detailed model of the BRDF based on the grammar of weaving at distinct locations

within the textile. Such studies can lead to a Bi-Directional Texture Function (BTF) or a spatially varying BRDF representation for the cloth (Adabala et al. 2003). Such approaches cannot be real-time and has application in off-line rendering of realistic textiles/clothes for digital production rather than for design where the real-time visualization is more useful. One of the important features of cloth that has not been captured is the presence of fibers on the surface of the cloth. We are investigating methods for overcoming this limitation.

ACKNOWLEDGEMENTS

The authors are grateful to Christiane Luible for providing the model of the clothes used in the Figures 5 to 7. Part of this work has been supported by the European project MELIES IST-2000-28700.

REFERENCES

- Adabala N, Magnenat-Thalmann N, Fei G, Visualization of Woven Cloth, to appear in *Proceedings of the Eurographics Symposium on Rendering 2003*
- Ashikhmin M, Premoze S, Shirley P (2000) A Microfacet Based BRDF Generator. In: *Proceedings of SIGGRAPH 2000*, pp 65-74.
- Baraff D and Witkin A (1998) Large steps in cloth simulation. In: *Proceedings of SIGGRAPH 1998*, pp 43--54.
- Clapp TG, Peng H, (1990) "Buckling of Woven Fabrics", *Textile Research Journal*, 60, pp 228-234 & 285-292.
- Collier JR, Collier BJ, O'Toole G, Sargand SM, (1991) "Drape Prediction by means of Finite-Element Analysis", *Journal of the Textile Institute*, 82 (1), pp 96-107.
- Cordier F and Magnenat-Thalmann N (2002) Real time animation of dresses virtual human. In *EUROGRAPHICS 2002, Computer Graphics Forum*, Blackwell Publishing, vol. 21, 327-336.
- Daubert K, Lensch HPA, Heindrich W, Seidel H-P (2001) Efficient Cloth Modelling and Rendering. In: *Eurographics Workshop on Rendering 2001*, pp 63-70.
- Gan L, et al, (1995) "A Study of Fabric Deformation using Non-Linear Finite Elements", *Textile Research Journal*, 65(11), pp 660-668.
- Gröller, E., Rau, R., And Strasser, W. (1995). Modelling and visualization of knitwear. In *IEEE Transactions on Visualization and Compute Graphics*, 302-310.
- Gröller, E., Rau, R., And Strasser, W. (1996). Modelling textile as three dimensional texture. In *Rendering Techniques 1996: Proc. 7th Eurographics Workshop on Rendering*, 205-214.
- Heidrich W, Daubert K, Kautz J and Seidel H-P (2000) Illuminating micro geometry based on precomputed visibility. In: *Proceedings of SIGGRAPH 2000*, pp.455-464.
- Kang TJ, Yu WR,(1995)"Drape Simulation of Woven Fabric Using the Finite-Element Method", *Journal of the Textile Institute*, 86(4), pp 635-648.
- Kautz J, McCool M (1999) Interactive Rendering with Arbitrary BRDFs using Separable Approximations, In: *Eurographics Workshop on Rendering 1999*, pp 281-292.
- Kawabata S, "The Standardization and Analysis of Hand Evaluation", *Hand Eval. and Stand. Committee of the Textile Machinery Society of Japan*, Osaka, 1975.
- Meissner M and Eberhardt B (1998) The art of knitted fabrics: Realistic and physically based modelling of knitted fabrics. *Computer Graphics Forum*, vol 17, no 3, pp 355-362.
- Nielsen R, Keates R, Sinkler R, Sally Breckenridge, Dana Cartwright, Jane Eisenstein, Mark Kloosterman, Bjorn Myhre. (1997) Weaving Information File Version 1.1. <http://www.mhsoft.com/wif/wif.html>.
- Sloan P-P, Cohen MF (2000) Interactive Horizon Mapping, In: *Eurographics Workshop on Rendering 2000* pp281-286.
- Volino P and Magnenat-Thalmann N (2000) Implementing fast Cloth Simulation with Collision Response. In: *Computer Graphics International 2000*, June 2000, pp257-268.
- Westin S, Arvo JR, Torrance KE (1992) Predicting Reflectance Functions from Complex Surfaces. In: *Proceedings of SIGGRAPH 1992*, pp 255-264.
- Xu Y, Chen Y, Lin S, Zhong H, Wu E, Guo B and Shum H (2001) Photorealistic Rendering of Knitwear Using the Lumislice. In: *Proceedings of SIGGRAPH 2001*, pp 391-398.
- Yasuda T, Yokoi S, Toriwaki J, Inagaki K (1992) A Shading Model for Cloth Objects. *IEEE Computer Graphics and Applications*, vol 12, no 6, pp 15-24.
- Zhong H, Xu Y, Guo B and Shum H (2000) Realistic and Efficient Rendering of Free-Form Knitwear, *Journal of Visualization and Computer Animation*, Special Issue on Cloth Simulation, 2000 pp13-22.

BIOGRAPHY

NEEHARIKA ADABALA obtained her Ph.D. from Indian Institute of Science, Bangalore in 2000. She worked at Philips Research, Bangalore, before joining MIRALab - University of Geneva as a post-doctoral research assistant. Her research at MIRALab focuses on realistic rendering of woven clothes.

NADIA MAGNENAT-THALMANN has pioneered research into virtual humans over the last 20 years. She studied at the University of Geneva and obtained several degrees including Psychology, Biology, Chemistry, and a PhD in Quantum Physics at the University of Geneva in 1977. From 1977 to 1989, she was a Professor at the University of Montreal in Canada. In 1989, she founded MIRALab, an interdisciplinary creative research laboratory at the University of Geneva. She has published more than 200 papers, is editor-in-chief of the Visual Computer and the Journal of Visualization and Computer Animation journal. She has received several awards and recently, she has been nominated at the Swiss Academy of Technical Sciences.

GUANGZHENG FEI obtained his Msc.degree from Hefei University of Technology in 1997, and obtained his Ph.D degree from Institute of Software, Chinese Academy of Sciences in 2001. After doing a short time research in Microsoft Research China, he worked in MIRALab, University of Geneva as a postdoc from 2001 to 2002 before moving to Animation School, Beijing Broadcasting Institute.

Virtual Textiles II

MODELING NEW PRODUCT DEVELOPMENT IN THE TEXTILE AND APPAREL INDUSTRY

Balamurugan S Ramanan, Master's Candidate in Integrated Manufacturing Systems Engineering
Dr. Padmini S Hands, Visiting Research Professor in Computer and Information Applications
Dr. Samuel C. Winchester, Klopman Distinguished Professor of Technology Management
College of Textiles, Campus Box 8301
North Carolina State University
Raleigh, NC 27606 USA
E-mail: bramana@unity.ncsu.edu

KEYWORDS

Management science, Decision-making, Management games, Model design, Parameter identification

ABSTRACT

This paper demonstrates a method to model the new product development (NPD) process, considering the apparel industry as an example case. Built as a multi-player simulation game, the value of this model is that it lets the decision makers in NPD evaluate the wide-ranging impact of their decisions and enables them to examine the effect of compromises involved in product development that inevitably occur in resolving conflicting pressures.

PROBLEMS IN NPD

New Product Development capabilities are increasingly becoming core competencies and key differentiators in mature industries (Prahalad and Hamel 1990) including the textile and apparel industry. Developing products that are attractive to the market within the available resources of development capacity, time and departmental budget has become a reality all managers are to live with. The NPD process has also become more complex and chaotic as more departments and external supply chain partners are involved early in the product development process, as customer requirements are more systematically sought, and as time-to-market becomes more critical.

Portfolio management and the prioritization of new product projects are critical management tasks in NPD (Cooper, Edgett, and Kleinschmidt 1998). This is often accomplished by a stage-gate process where, the projects go through cycles of approval and are either approved for further development or are 'killed'. This approach to innovation creates redundancy and variety, which in turn allow flexible responses to new circumstances (Levinthal and March 1993; Levitt and March 1988). But the, NPD process is also becoming increasingly an offer-based' process where project

selection is more 'choosing the best among the available alternatives' than being specification driven.

Another problem associated with the stage-gate process is the level of risk taking in the decision making process. NPD in apparel industry suffers from an excess of Type I errors where the development lines are clogged with an excess of projects because the management is averse to stopping any project that might be successful and postpones a 'kill' decision as long as possible. This leads to self-inflicted resource constraints and increased development costs. This is also because of the tremendous pressure on companies to be in a position to respond immediately to external stimuli (from the market, i.e. customers and competitors). Industries in this situation, such as textile and apparels that are linked tightly to consumers typically keep all product options open until the moment of market entry, stretching available resources to the maximum.

There is also the constant tension between 'lead' and 'follow' business strategies. Research shows that long-term competitive viability stems from both effective exploitation of existing knowledge and the exploration and development of new knowledge (Freeman 1982; Porter 1991; Damanpour 1991). The "follow" strategy is more deterministic as customer conditions are clear, and there is the operational efficiency of not developing multiple options. At the same time, companies realize the need to take risks and be the first to introduce big winners in the market, in order to gain market share even temporarily. When clear communication is not available from company leadership as to which strategy is pertinent at any given point, this leads to tension and confusion among the NPD players about the rationale behind business decisions. With declining profitability and increasing development costs, there is also a pressure to be operationally efficient and invest only in highly likely winners.

With these multiple and complex problems, it is difficult to evaluate the impact of decisions and the rightness or wrongness of individual decisions. Business success or

failure of a project is often the only metric for the decision process.

NEED FOR A TRAINING SIMULATOR

There is clear distinction between a *decision having the right outcome* and the right *decision process*. An incorrect decision may have successful outcome because of factors not in the control of the decision makers. NPD managers should practice and learn strategies to make good decisions even in the absence of clear feedback for their decisions.

A simulation game shows clear 'cause and effect relationship' between the decisions and their outcome by reducing complexity, providing real-time feedback, and by providing a virtual world of exaggerated reality.

Objectives of WISsion

This paper demonstrates WISsion, a training simulator aimed to determine the right decision process in different environments. The training simulator WISsion, a term derived from 'Wise decision' and rhyming with 'Vision' enables the users to experience the realities and challenges of NPD process so that they can

- Experience the conflicting pressures of resource efficiency and design excellence
- Practice taking decisions and making compromises
- Better understand how postponing decisions affects the development pipeline
- Understand the relationship between local measures and global performance

In particular, it allows the team of users to experiment with various injections into realistic product development processes and to evaluate their effectiveness regarding how NPD can be better managed by providing a real time feedback on the impact of their decisions on:

- When to start projects with respect to expected market launch dates?
- When to kill projects?
- How to set priorities for scheduling?
- How to evaluate the likelihood of success and the associated confidence level in our measurement
- How to evaluate resource utilization in NPD (shape of development cone – rate of whittling down projects from design stage to market introduction stage)

The objective of WISsion is neither to develop an in-depth understanding of textile industry or any decision support systems. WISsion uses textile and apparel industry only as an example domain and the users do not need prior experience in textile industry to use WISsion. It also does not require extensive knowledge or market analysis tools or decision support systems. All the information needed to take decisions is provided in the form of easy-to-interpret charts.

Challenges in creating a NPD Simulator

Product development management can be considered to occur at two levels. Product level decisions determine whether a product or a line of products should be continued. These can be viewed as portfolio or product management decisions. Process management decisions within each of these processes determine the priority, scheduling, resource allocation and quality standard for that work. The existing simulation packages either concentrate at the product level decision-making (marketing and product management) or on operational and resource management issues (Manufacturing and supply chain simulations). But NPD is as much about product decisions as about project/process management. WISsion integrates these two views by simulating both the product management and operational level decisions. Another problem in simulating NPD is translation of subjective data, such as customer preference and offer price, to pieces of information that are both simulatable and user-friendly. In WISsion, long strings of numbers are used to denote each of these parameters. However this information is always presented to the user in a graphical format where it is compared with another parameter. This relative display lets users understand the relative goodness of the decisions and aids them in changing decision strategies to improve performance.

Further there is the challenge of representing a portfolio of offers and their associated market volume and profitability.

MODELING WISsion

Modeling Philosophy & Theory of Constraints Applied to Product Development

The Theory of Constraints Thinking Process (TOC TP) approach has been successfully applied to various streams of management including scheduling and production management, project management and accounting with highly successful results (Goldratt). It was decided to apply the Theory of Constraints thinking process to the New Product Development process. The process was initiated by collecting the chief irritants of the key NPD players. TOC calls these as the undesirable effects (UDEs).

Theory of Constraints Thinking process approach to problem solving believes that an UDE exists in the organization because the organization is caught between conflicting needs and as the balance of the organizational policy or practice tilts towards one of these conflicting needs, we feel an undesirable effect from the other conflicting need. TOC indicates that there is ONE core conflict in the organization and that it is manifested as different conflicts in day-to-day operations. The various UDEs are only symptoms of this core conflict.

Core Conflict in NPD

Application of TOC to the NPD process of textile and apparel industry revealed that the core conflict in the NPD is the conflicting pressure between the need to be responsive to market needs and the need to operationally efficient. NPD

players believe that, in order to have a successful NPD, they should be in a position to respond immediately to the needs of the market and in order to be in a position to respond to market fast, they invest their resources in agility by keeping their options open. On the other hand they also believe that in order to have a successful NPD they should be operationally efficient and so feel the need to reduce our development costs. The need to invest in agility and the need to cut costs are in direct conflict with each other. This core conflict is represented pictorially in FIGURE 1.

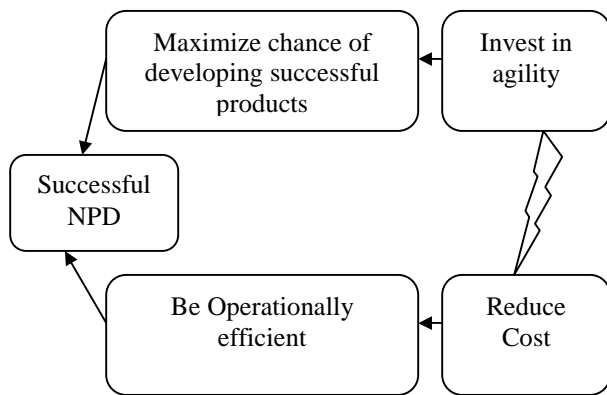


Figure 1: Core conflict in NPD

The conflicting pressure on NPD managers to maintain operational efficiency and at the same time remain responsive to market changes frequently leads to conflicting strategies and compromised decision-making process. For example, the need for additional relevant information is ignored because of the cost of information or the lack of time leading to a possible missed market opportunity. Decision-making under great uncertainty than necessary has become the norm. Additionally, the point of view is that we are satisfying, rather than optimizing to known conditions.

This conflict between operational efficiency and the need to cater to the market is the fundamental building block of the training simulator. The simulator is built based on the principle that market prefers an ideal product at an ideal price and as the separation between these and the offer increases the preference for the product decreases. It is nearly impossible to accurately predict market preferences as they change with time and as the resource requirements to do that would be too high. The goal of the players is to achieve a balance between achieving a perfect match with the market preferences and utilizing the resources effectively.

The product development process can broadly be said to comprise the following phases: understanding the market, understanding market needs, product development, manufacturing and market entry decisions. In each of these phase, resource is spent to create information that will aid the decision making process. But the available resources are to be

apportioned between a portfolio of products that are to be developed and there is always a compromise between the quality of work and resource realities. At some points the players decide the level of information to be an 'accurate enough approximation' and pass it on to the next stage of development. When further development work is based on this approximation, the margins of error are increased and the end product is separated from the market-desired product. While this separation affects customer preference and affects the revenue generation potential of individual products, on a global scale it provides the NPD department with enough resources to manage a portfolio of products.

WISSion is based on this philosophy of compromise in resource management. FIGURE 2 is a pictorial representation of the modeling philosophy used.

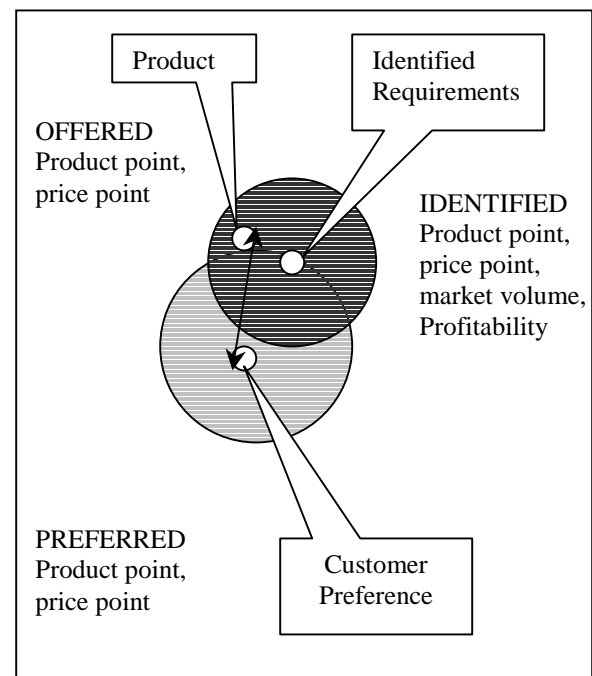


Figure 2: Modeling Philosophy

Key Players

The key players in the NPD process are merchandising, marketing, design & engineering, and manufacturing & tooling departments. These departments strive to achieve business match, product match, price match and a profitable production to maximize their revenues, profitability and share in market. They achieve this by working on separate pieces of information based on the inputs they receive from others and then combining them to aid the decision-making process. Apart from the broad objective of creating products that sell, they are also responsible for the operational efficiency of their departments. The game is constructed to simulate the apparel industry working to multiple seasons per year. The

game is played as a series of seasons of product development. The decisions taken at any point have a short-term and a long-term impact.

Decisions

Typical decisions taken in the game include choosing a project that has attractive volume and profitability potential, deciding the level of accuracy to which work is to be completed, deciding to continue or kill a project and decision on the pricing strategy. Some of these decisions are direct yes / no type decisions, while others are either fed to the simulation engine as input parameters for work or are indirect variables which are driven by the effect of other decisions and thus cannot be controlled directly. For example, profit margin cannot be controlled directly as it is dependent on the cost structure and the pricing decisions. Following table summarizes the key components of action in WISSion.

Table 1 Responsibility Matrix

	Market Knowledge	Product Knowledge	Price Match	Manufacturing
Merchandising	Marketing budget	Product launch approval	Pricing	
Marketing	Market volume, profit	Product Knowledge	Price point	
Design		Product Design	Material cost	Manufacturability
Manufacturing			Manufacturing cost	Faithfulness, repeatability

Decisions are taken with respect to each of these activities. As can be seen in Table 1, teamwork is essential for each NPD objectives to be fulfilled as different partners work on different pieces of activity for the same NPD objectives. Also the quality of work at the upstream level affects the quality level at the down-stream level. For example, when manufacturability of a product as created by design is low, the faithfulness and repeatability are also low and manufacturing has to spend extra resources to correct the initial problems the project may have to be killed.

Performance Metrics

Two sets of performance metrics measure performance in the game - functional level performance measures and company level performance measures. Functional level performance measures include, productivity, efficiency, average turn-around time, overhead costs, resources spent on rework etc., Company level performance measures report the NPD performance of the total company. This includes profitability, market share, number of products introduced per season, revenue growth and number of projects developed from one stage to the next (development cone).

GAME LOGIC

Architecture

WISSion is built using Arena 5.00.02 (© Rockwell Software Inc.) as the simulation engine and MS Excel as the user-interface. All information from the simulation engine is made available to the players through a series of user forms in MS Excel. In the first prototype, networking between the five computers is done by using the networking facilities available in Windows NT.

All information is provided to the players as a series of charts. Based on these information the players take operational and product level decisions. These decisions in turn change the information and also affect the performance at both the global and the functional level. The key information available to the users and the main actions taken by them are represented in FIGURE III

Views And Actions

At the start of the simulation, the user is taken to an 'in-tray' screen where all the projects that need his/her attention are displayed. The information displayed in the form include, project number, the season it is meant to be released in market, lead or follow mode of business, information on whether it is a rework, inspection of a previously completed job or new job from his colleagues. The user can sort his in-tray by any of these parameters and choose to work on any of the projects. This screen takes the user to the inspection screen where the player can gain information on the portfolio, cost structure and manufacturability.

The inspection screen is the primary screen for decision-making. It carries the three basic views of product and the entry points for all actions. The portfolio view carries information about the market size, intended season, profitability and completion status of projects. This option can be used to decide the relative attractiveness of business decisions and is primarily used by the players representing merchandising department. FIGURE 3 gives the screen capture of the portfolio view.

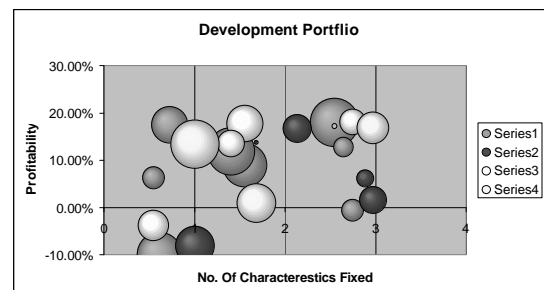


Figure 3: Portfolio View

The cost structure view compares the market preferred price point and estimation for the developed product. Cost is broken down as material cost, manufacturing cost, quality

cost, amortized development cost, overhead cost, marketing & advertising cost and profit margin. FIGURE 4 gives the cost structure view. As the estimated costs are provided relative to an 'estimated ideal' cost, the players deal only with the relative cost elements but not with absolute elements. This also eliminates the possibility of the players getting confused by possible differences in cost among different products.

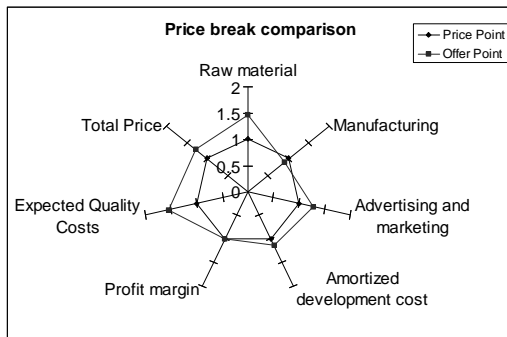


Figure 4: Cost Structure View

The manufacturability view is also a comparison chart between the developed product and an ideal manufacturable product. This view provides details about manufacturability, faithfulness, responsiveness and manufacturing cost. FIGURE 5 is an example of manufacturability view.

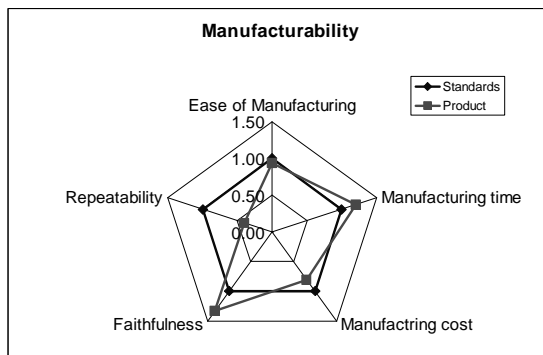


Figure 5: Manufacturability View

Apart from viewing the portfolio, cost structure or manufacturability the player can send jobs to other departments, kill a project, proceed to introduce the product in market or schedule work in his department. Players can also request the other players to complete the job with a higher priority and add comments.

In the 'Work Status and Accuracy View' of FIGURE 6, the user can visually evaluate the work on creating an NPD offer with respect to information available regarding market need. The product offer and the market understanding are represented by two circles. The diameter and position in the space of the two circles are significant. Size indicates precision - both of market understanding and the incomplete product offer. Position is cardinal and even ordinal

meaningless, since product characteristics and market preference are in an undefined n-space. However, the distance between the two circles measures the distance between market preference and the product offer.

The work status and accuracy view can be used for any of the NPD Objectives - Product match, price match, market knowledge or manufacturability. FIGURE 6 is an example of the work status and accuracy view where the shaded circle represents the accuracy of market information and the outer circle precision of the offer.

In the case of new to the world products, some information such as preferred price would not be available for use in decision-making. The team is expected to evolve a strategy to handle these projects in the absence of information. The team can choose what proportion of the NPD portfolio will involve new to the world products.

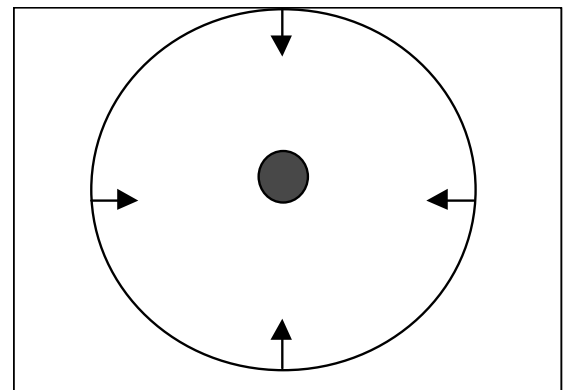


Figure 6: Work Status and Accuracy View

In the 'Work Scheduler Form' [not shown], which also can be reached from the inspection screen form, work can be scheduled for any of three functions – marketing, design & engineering or manufacturing and tooling. Irrespective of what functional area the work is done on, the resources that are available with the player (simulated workers) will be used to do this work. As the manager for their functions, players can specify what aspects of a particular function they want their team to work on, the level of accuracy desired and whether or not to freeze specification for a particular characteristic. By doing more work on a particular aspect, the player either increases the accuracy of information or the precision of the offer with respect to the information available. This translates to reducing the size of work status or accuracy circles and increasing the probability of matching more precisely with market preference.

When the players feel that they have achieved a reasonable precision they can choose to freeze the specification by selecting an arbitrary point within the circle. Alternately, they can leave the specification open. By not freezing the Specifications, resource intensive reworks can be eliminated, but specifications have to be finalized before making the

In the 'Market Introduction View' [not shown], decision regarding the level of marketing and advertising needed for the product and the pricing strategy for the product is decided.

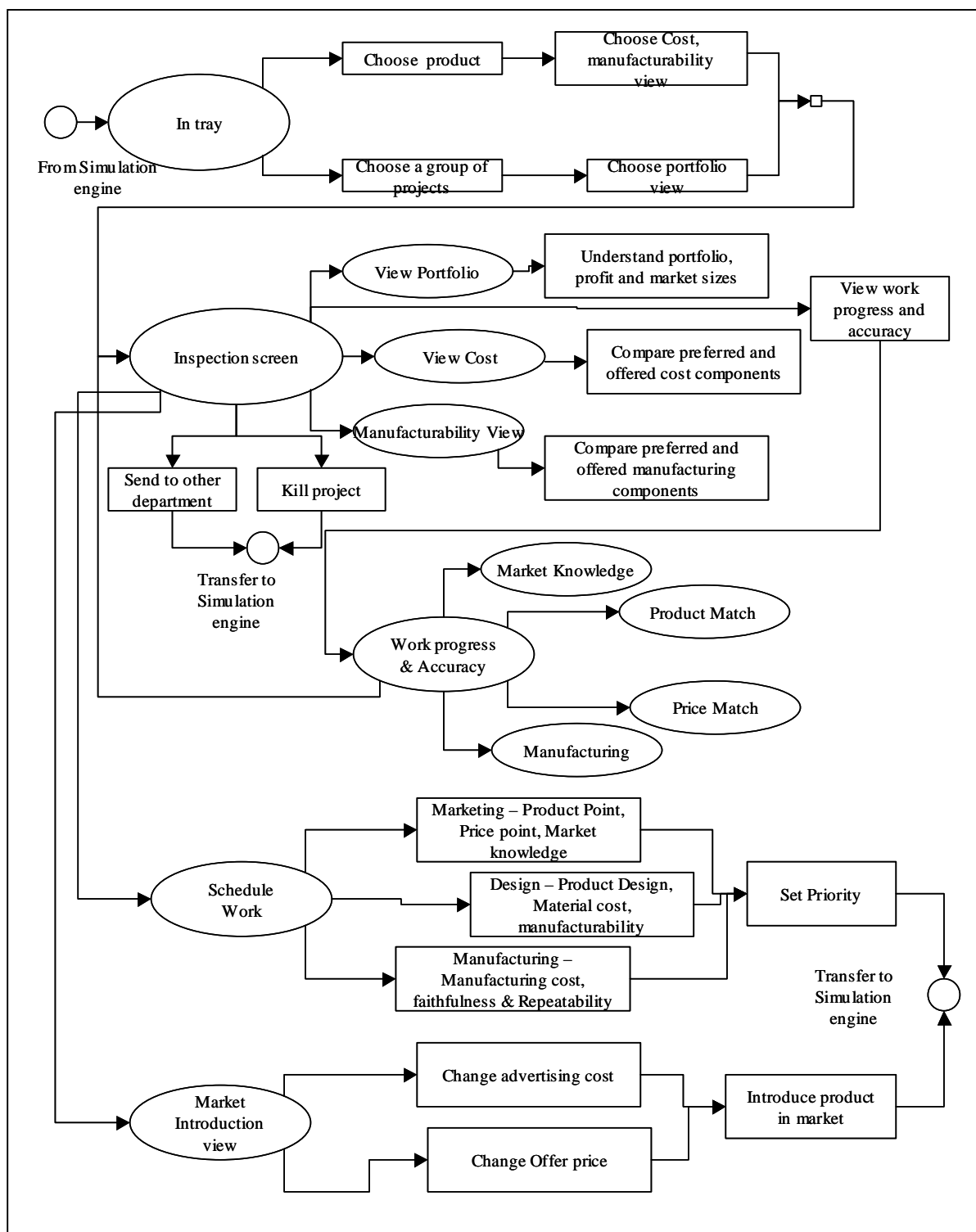


Figure 7: Game Logic for ‘Apparel NPD Simulator’ - Views and Actions (Petri-net model of user screens in the

By increasing the advertising spend or by reducing the price of the product, the relative attractiveness of the product can be increased, but profitability would be reduced because of the increased cost. FIGURE VII is a Petri-net diagram of the overall game logic where the 'views' are represented by ellipses and the actionable items are represented as rectangles.

On market introduction, the preference points for a product (product, price and time of introduction), which have been fluctuating with respect to time, are matched with the offer points. Depending on the separation between preference and offer, the success of the product is determined. This is reflected as sales volume. Revenue generation and profitability are determined by the relationship between preferred price, cost structure, pricing strategy and sales volume. All metrics for a season are consolidated and displayed as performance of the company or the function for that particular season.

MODELING THE REALITIES OF PRODUCT DEVELOPMENT

This section explains how the training simulator captures the realities of the product development environment and the modeling techniques used to achieve this. The basic technique employed in modeling 'WISsion' is using multi-digit integers to represent the parameters, and sampling them from the most significant digit onwards, to represent increasing levels of precision or certainty. Shifts in these parameters over time (for example changes in market preference) are modeled as changes to the least significant digits as the effect of time. Such a system, effectively translates decreasing vagueness or uncertainty in product characteristics and market understanding, and allows us to represent subjective and qualitative information such as market preference for a color, as simulatable quantities. By knowing a series of digits from the most significant digit, the set of digits that are not yet determined dictate the size of the information or work completion circle. All points within the circle have equal probability of being the product or the offer point.

Modeling Cost Reduction

The highest cost associated with a circle is assumed to be the representative cost for a group of points bound by an accuracy circle. So, the actual cost on arbitrarily choosing a point could be lower than the predicted cost. Cost reduction is achieved by reducing the size of accuracy circles (by investing more time) and thereby reducing both the maximum cost and the probability of choosing a higher cost.

Modeling Effect of Time

With progression of time, the market preferences change. These changes are added to the least significant digit of the simulation parameter. Over time, these random changes start impacting more significant digits and thus create a trend. At

any point in time, the preference points lie in the base circle of a time cone the height of which is given by the time lag since measuring a particular parameter. The level of fluctuation determines the half angle of the development cone. By reducing the development time, the players maximize the chance of reducing the separation between the measured value of the parameter and the current value of parameter.

Time of entry in the market

All products and their markets are higher for a particular season and wanes in other seasons. Late entrants for a season lose part of the sale in proportion to their lateness. This is modeled by modeling the sales volume on a per-day basis. On seasons other than the season the product is meant for, the per-day volume for the product is reduced. This reduction is a random process.

New-to- World Products

The simulation engine circulates the market segments with a built in randomness. While some segments are retired other new segments are generated. These new segments do not carry information about their product point or price point trends. The simulation team works on these products without this information. As in the real world, the new-to-world products carry a higher risk but have higher returns. This is modeled by increasing the profitability of these segments and the time required to increase the accuracy level of information.

Modeling the relationships between factors

The parameters of the governing equation and the relationship between the equations determine the validity of any simulation experiment. But the relationship between most variables in this simulation game is empirical at maximum. This is an area of continuing research activities and in a sense even this project is an attempt at determining the relationship between various parameters. So, a best-case approximation of relationships based on the textile industry supply chain business model (Demand Activated Manufacturing Project) is used. FIGURE 8 gives the relationship between the simulation variables. This relationship proceeds from simulation level activities by individual players to their impact on product match, price match and sales volume. Performance metrics are derived from these relationships.

Tailoring the training simulator trainer to various environments- To maintain flexibility and make the simulator applicable to varying circumstances, parameter control is given to the users. Depending on the level of competitive intensity in their market and the stability of their economy, users can set simulation control parameters such as price fluctuation, demand stability, price tolerance etc. The base levels are set based on the economic and industry indicators. For example, variation of price is modeled on

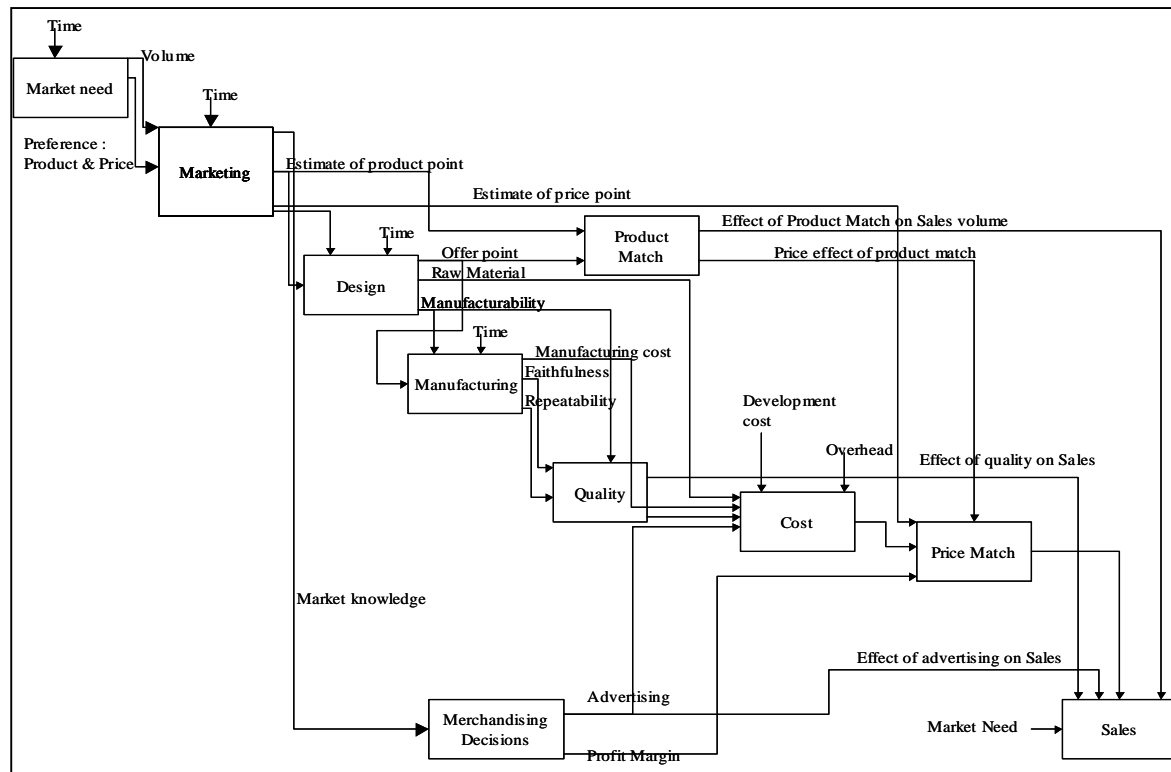


Figure 8: Relationship Between Simulation Variables and Performance Metrics (IDEF 0 Modeling)

twenty-year inflation figures and market volume is modeled on the apparel industry growth trend over the last ten years.

CONCLUSIONS & PATH FORWARD

The proposed simulator provides a creative way to model the NPD processes. The value of the simulator is in letting the training groups realize that managing the conflict between operational efficiency and customer satisfaction is a reality in product development and derive a strategy to handle that. It is a good tool not only for training NPD personnel but can also be used as a research tool for observing behavioral patterns, risk preference and coordination skills of various groups. This can also be used to determine the preferred set of performance metrics to reflect actual success in a given market and organizational environment.

WISsion can further be enhanced by creating multi-team interfaces and allowing multiple teams to compete with each other. A messaging tool like Yahoo! Messenger (© Yahoo! Inc.,) can be incorporated with the Simulator.

That will facilitate communication between the team members and possibly enhance decision-making. In the present version, the simulator needs a minimum of four people to play the game. By automating some of the decisions, the game can be converted to a single-user game,

which can be used for on-the job training. Integrating the simulator with an intelligent agent will allow it to learn from the experience of all previous games.

REFERENCES

- Cooper, R. G., S.J. Edgett, E.J. Kleinschmidt (1998), *Portfolio Management for New Products*, Reading, MA: Perseus Books.
- Demand Activated Manufacturing Architecture (DAMA project), Textile / Clothing technology corporation, 211, Gresson Drive, Cary NC 27511
- Leonard-Barton, D. Freeman, C. (1982), *The Economics of Industrial Innovation*, Cambridge, MA: MIT Press.
- Levinthal, D. and J.G. March (1993), "The Myopia of Learning," *Strategic Management Journal*, 14, 95-112.
- Levitt, B. and J. March (1988), "Organizational Learning," *Annual Review of Sociology*, 14, 319-340.
- Porter, M.E. (1991), "Towards a Dynamic Theory of Strategy," *Strategic Management Journal*, 12, 95-117.
- Prahalad, C.K. and G. Hamel (1990), "The Core Competence of the Corporation," *Harvard Business Review*, May/June, 79-91, (1995)
- Wellsprings of Knowledge: Building and Sustaining the Sources of Innovation, Boston, MA: Harvard Business Press.

CAD/CAM for Cellular Woven Fabrics with Trapezoid Cross-sectional Shapes

Xiaogang Chen, Hongxia Wang, and Hongxia Zhang
Department of Textiles
UMIST, PO Box 88, Sackville Street
Manchester, UK M60 1QD
Email: Xiaogang.Chen@umist.ac.uk

KEYWORDS

CAD/CAM, cellular fabrics, mathematical modelling

ABSTRACT

This paper introduces the design and implementation of the CAD/CAM software for multi-layer cellular woven structures with trapezoid cross-sectional shape. It is concentrated on the establishment of mathematical models for cellular woven structures with trapezoid cross-sectional shape. A lifting plan described as a 2D binary matrix can be automatically created after the specific parameters of structure and weaves are defined. NURBS is used for the visualization of multi-layer cellular woven fabrics.

INTRODUCTION

Multi-layer cellular woven structures are of potential uses for cushions and energy absorption because of their light-weight and inflatable volume. An advantage of such structures is that they can be produced on conventional looms.

Cellular woven structures are composed of multiple sets of warp and weft yarns. The design process of these structures is complicated and time consuming and is very technical and therefore requires high level of concentration. There are a variety of CAD programs available for the design and production of woven fabrics, but none of them are suitable for the design and production of cellular woven structures. A CAD/CAM software package is developed at UMIST that helps engineers to design cellular woven structures by simply specifying structural parameters.

This paper mainly describes the planning and implementation of the CAD/CAM software for cellular woven fabrics.

DESIGN OF CELLULAR STRUCTURES

Cellular fabrics with trapezoid cross-sectional shape

Takenaka and Sato have patented a cellular woven structure with honeycomb cross-sectional shape (1991). All layers have the same length and the adjacent layers of the fabric are woven together according to the designed cross-sectional shape. A CAD/CAM software developed at UMIST deals with the design and manufacture of this type of structure. However, the

surface of this type of structure is uneven, which can limit its application in some cases.

Figure 1 shows a different type of cellular woven structure with trapezoid cross-sectional shape. One trapezoid is defined as a cell. Each solid line segment (side of trapezoid) in Figure 1 represents a section of fabric.

The characteristics of this structure are:

1. It has flat top and bottom surfaces;
2. Each independent layer has different length;
3. Trapezoid cross-sectional shape is very flexible and can be changed into a triangle or rectangle if adjusting the angle and length of the sides, as shown in Figure 1(a);
4. Different structures can be obtained with multiple levels of trapezoid being used, as shown in Figure 1(b);
5. It allows many levels, only limited by the loom capacity, of cells and different sizes and shapes of cells to provide the required energy absorbency and thickness.

Structure modelling

This type of cellular structure consists of an array of cells with trapezoid as its basic shape. Each cell is defined by the lengths of its sides and the angle between the sides. Cells are arranged in certain rows and columns to construct a cellular structure. The structural shape will be different if changing relative position of the rows. So a cellular structural model can be defined by the following parameters that are shown in Figure 2.

A, B, C, D : the lengths of trapezoid sides, measured in number of picks

α : the expansion angle

Rp_i : the relative position between the row i and the row $i+1$

Obviously, $B > A \sin \alpha$.

Rp_i can be of any reasonable value. However there are too many varieties of this structure. Some of these are not of any practical use. Two cases have been employed in this CAD/CAM software as shown in Figure 1(b).

Case 1: $Rp_1 = Rp_2 = \dots = Rp_{n-1} = 0$ (top)

Case 2: $Rp_1 = Rp_2 = \dots = Rp_{n-1} = A \cos \alpha + C$ (bottom)

where n is the number of levels (rows) in cellular structure. This paper mainly discusses the case 1.

A side is the minimum unit in this structure, and each side is a section of fabric. Different weaves can be assigned to different fabric section. X. Chen *et al.* (1996) have created the formula method for inputting weave parameters in order to create weaves. For regular weave, the parameters include the weave formula which is a series of numbers that indicate the lengths of floats and the weave step number.

CREATION OF LIFTING PLAN

The set of instruction codes that controls the shedding mechanism is known as the lifting plan and it can be created from the weave and the draft. In cellular structures, once the fabrics weave is defined, the lifting plan for weaving the fabric structure can then be generated automatically. To represent a weave mathematically, a 2D matrix is used to record the data of a weave (Newton *et al.*, 1979). Chen *et al.* (1996) have explained the mathematical representations of weaves in detail. The intention of this part is to discuss how to employ this technique to generate lifting plan for the cellular structures with trapezoid cross-sectional shape.

The single-layer and stitched-layer weaves

For a single-layer fabrics weave, after the weave parameters are defined, a 2D binary matrix can be generated which represents a weave mathematically. Its elements are either 0 or 1. '1' indicates a warp-over-weft crossover, and '0' means a weft-over-warp crossover. The position of each element in the matrix is located by a pair of co-ordinates (x, y) where x indicates the x th column from the left and y indicates the y th row from the bottom.

Each layer of cellular structure is relatively independent from others. Parts of adjacent layers are stitched together. These layers are called stitched-layers in this paper. There are various ways of stitching. A valid stitching method is one that does not interfere with the fabric structure.

One valid method is employed in this CAD/CAM software. Figure 3 illustrates the cross-sectional view of 3 layers cellular structure using this stitching method. The warp ends from lower (upper) layer go along with the corresponding warp ends from upper (lower) layer. The matrix of this stitched-layer is just two copies of upper (lower) layer 2D binary matrix arranged together by column. If the dimension of the upper (lower) layer matrix is $m \times n$, the dimension of the stitched-layer matrix is $m \times 2n$.

Cellular structure subdivision

In order to generate the lifting plan for the whole cellular structure, the repeat unit of this cellular structure is determined by some kind of regular pattern. Figure 4 shows an example of a three-layer cellular structure with three repeat units, and the rest of the discussion in this paper will be based on this structure.

However, it is not easy to generate the lifting plan for one whole repeat. Therefore, one repeat is subdivided into four areas as shown in Figure 4. The details are discussed below.

Area 1 includes three sections of fabrics. These sections have different lengths and possibly different weaves. All sections are single-layers. The 2D weave matrix for this area can be generated by combining these three layers together.

Because all three layers are woven in one operation and they have different lengths, the weft yarn distribution for the three layers must be considered. The key issue here is that the weft yarn distribution must satisfy the definition of the trapezoid. According to the geometrical description of the cellular structure, the length ratio of these three sides in Area 1 is $[\cos(\alpha) : 1 : \cos(\alpha)]$. Since the lengths are measured by number of picks, the parameters of weft yarn distribution should be integers and may involve the use of multiple groups of weft yarn distribution. The number of groups is denoted by k . For the m layers cellular structure, the group i of weft yarn distribution is defined as: $w_{1i} : w_{2i} : w_{3i} : \dots : w_{mi}$ where all w_{ri} are integers ($r=1, 2, \dots, m$ and $i=1, 2, \dots, k$).

$$\sum_{i=1}^k w_{1i} : \sum_{i=1}^k w_{2i} : \sum_{i=1}^k w_{3i} : \sum_{i=1}^k w_{4i} : \dots : \sum_{i=1}^k w_{ri} \approx \cos \alpha : 1 : \cos \alpha : 1 : \dots : \cos \alpha$$

$$\text{In particular, } A = \sum_{i=1}^k w_{2i}.$$

This CAD/CAM software employs an algorithm to calculate a valid weft yarn distribution. The w_{ri} is calculated as follows:

$$w_{ri} = \begin{cases} 1 & \text{if } r \text{ is odd} \\ \text{round}(i / \cos \alpha - \sum_{j=1}^{i-1} w_{rj}) & \text{if } r \text{ is even} \end{cases}$$

For example, for the three layers cellular structure shown in Figure 4, $A=12$, $\alpha=60^\circ$, $\cos \alpha=1/2$.

$$\begin{aligned} w_{11}=1, w_{21}=\text{round}(1/\cos \alpha)=2, w_{31}=1; \\ w_{12}=1, w_{22}=\text{round}(2/\cos \alpha - w_{21})=2, w_{32}=1; \\ w_{13}=1, w_{23}=\text{round}(3/\cos \alpha - w_{21} - w_{22})=2, w_{33}=1; \\ \text{and so on.} \end{aligned}$$

So the weft yarn distribution $w_{1i} : w_{2i} : w_{3i}$ is $1 : 2 : 1$ where $i=1, \dots, k$ and $k=6$.

Since the three layers in Area 1 have same widths, warp yarn distribution is not considered at this stage in order to simplify the matrix operation.

Three 2D binary matrices which represent the three single-layer weaves can be combined together to generate the overall matrix for Area 1 using the method described by Chen *et al.* (1998).

Area 2 includes two sections of fabrics, each having the same lengths and possibly different weaves. The bottom section is a single-layer weave. The other one is a stitched-layer weave. The 2D weave matrix for this area can be generated by combining these two layers together.

Area 3 is similar to Area 1 and Area 4 is similar to Area 2.

Generating the lifting plan for whole structure

After the weaves of all single-layers and stitched-layers and the weft yarn distribution are defined, the 2D weave matrices for individual areas can be generated automatically in this CAD/CAM software package. One repeat of this structure is divided into four areas. The lifting plan for the whole repeat is the combination of four 2D matrices of four areas. One repeat lifting plan for the three layers cellular structure (Figure 4) is shown in Figure 5. The specific parameters are as follows:

$A = B = 4$ picks
 $C = D = 2$ picks
 angle = 60°
 formula = '1/1' (the plain weave)
 step number = 1
 weft yarn distribution = 1:2:1

In Figure 5, the vertical spaces represent the shafts and the horizontal ones the picks. A filled square indicates that this particular shaft will be lifted on a certain pick.

3D VISUALIZATION

The 3D woven fabric model gives a clear view of micro-structure of the fabric. This CAD/CAM software creates and displays the multi-layer cellular structures in three dimensions based on Shannan and Harle's lenticular model (1978). The solid 3D yarn model are generated from two Non-Uniform Rational B-Spline (NURBS) curves: shape curve and path curve (Piegl and Tiller, 1995), and this is illustrated in Figure 6. The curve representing the yarn path for lenticular fabric consists of two arcs and straight line. We offset one arc with the half height of the lenticular arcs, and get a new arc as part of the path. According to the weave pattern, the adjacent arcs are connected to generate the path curve. For the multi-layer cellular structure, 3D view of every fabric section and every area can be displayed. Figure 7 shows a single-layer fabric defined by the formula 1/1. Figure 8 shows 3D view of three layers fabrics in Area 1 (all 1/1) of a cellular structure.

CONCLUSION

The CAD/CAM software package is developed based on the above analysis and discussion. It is developed under Microsoft Windows environment. The development tool is Visual C++ and OpenGL. The main functions of this software include:

- The definition of cellular structure with trapezoid cross-sectional shape and weave parameters;

- The automatical generation of lifting plan;
- The visualization of 3D solid models for cellular structure fabrics.

Its friendly and easy-to-use interface displays four views. These are the model view for cellular structures (main windows), where the designers and engineers can easily define the parameters of cellular structure and weave, the lifting plan view, the 3D visualization view, and the 2D mesh view. Its file opening, saving and printing facilities give the designers and engineers great convenience in the process of fabrics designing and producing.

REFERENCES

- Takenaka, K, Sato E, Woven fabric having multilayer structure and composite structure comprising the woven fabric, US patent No. 5021283, 1991.
- Chen, X., Knox, R. T., McKenna, D.F., and Mather, R.R., Automatic Generation of Weaves for CAM of 2D and 3D Woven Textiles Structures. J. Text. Inst., 87, Part I, 356-370, 1996.
- Newton, A., and Sarkar, B. P., An Analysis of compound Weaves, J. Text. Inst., 70, 427-438, 1979.
- Chen, X., P.Potoyaraj, CAD/CAM for complex Woven Fabrics Part II: multilayer Fabric Automatic Generation of Weaves for CAM of 2D and 3D Woven Textiles Structures. J. Text. Inst., 87, Part I, 356-370, 1998.
- W.J.Shanahan and J.W.S.Hearle, An Energy Method for Calculation in Fabric Mechanics Part II: Examples of Application of the Method to Woven Fabrics. J.Text. Inst., 69, 92-100, 1978.
- Piegl L., Tiller W., The NURBS book, Springer-Verlag, 1995.

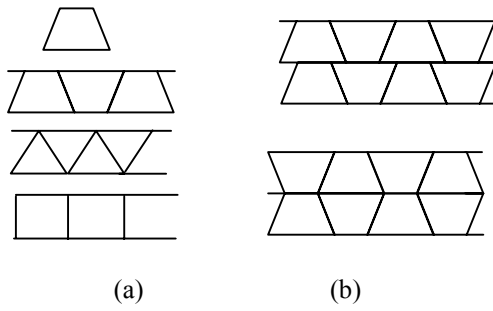


Figure 1 Variety of cellular woven structures with trapezoid cross-sectional shapes

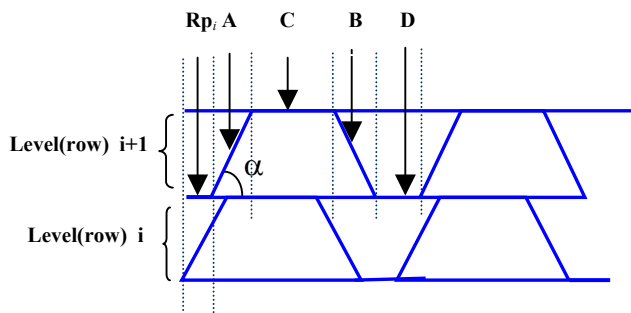


Figure 2 Geometrical description of cellular structure

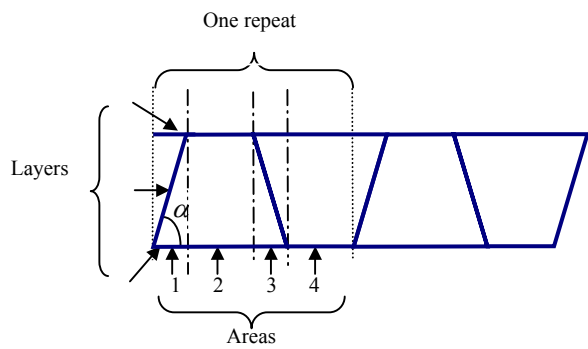


Figure 3 The cross-sectional view of 3-layer cellular structure

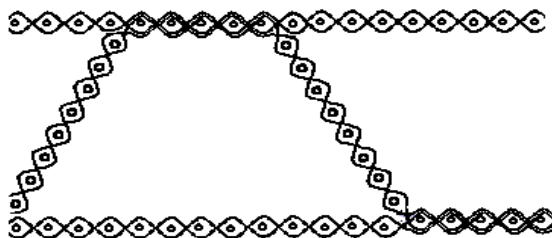


Figure 4 Cellular Structures Subdivision

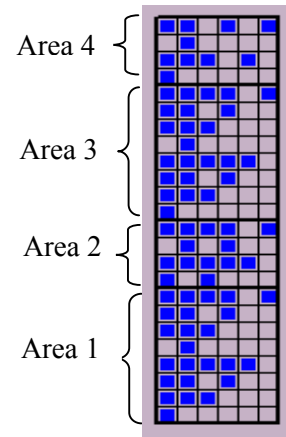


Figure 5 Lifting plan of three layers cellular structure

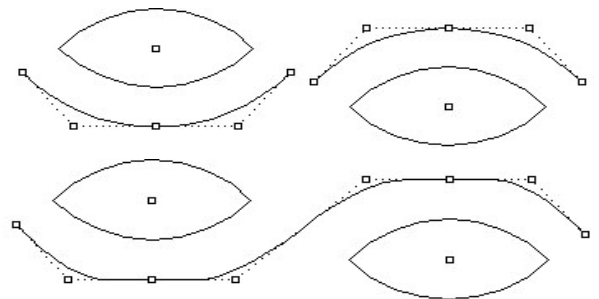


Figure 6 Designing the shape and the path curve

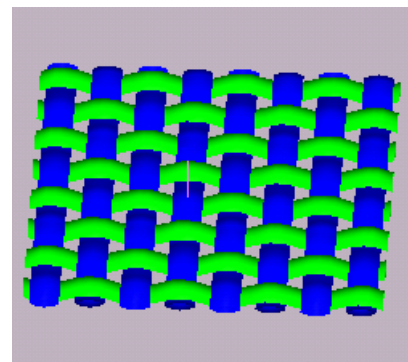


Figure 7 Single layer fabrics

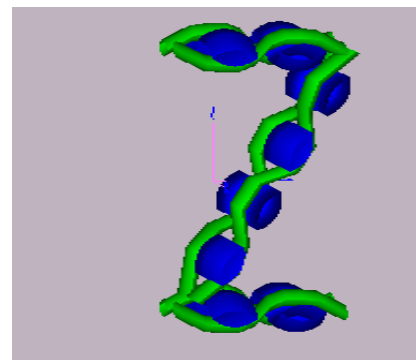


Figure 8 Fabrics in Area 1

TRACTION MODEL - A METHODOLOGICAL APPROACH

Pascal Bruniaux and Jean Marie Castelain
Génie et Matériaux Textiles (GEMTEX) Laboratory,
Ecole Nationale Supérieure des Arts et Industries Textiles
9, rue de l'Ermitage - BP 30329 - 59056 ROUBAIX Cedex 01
France

E-mail : pascal.bruniaux@ensait.fr - jean-marie.castelain@ensait.fr

KEYWORDS

Stretching, Stress-strain curve, Nonlinear model, Yarn modeling.

ABSTRACT

Paper presents the modeling of the phenomena of stretching, which becomes integrated into the model of yarn elaborated the team "modeling and simulation of structures textiles" of the laboratory GEMTEX. The analysis of the stress-strain curve of the yarn that we set up in our model, tried to interpret this curve in the form of various models. This differentiation is due to the change of state of the yarn at the time of its stretching. So, the model differs very clearly from the other models exposed in the literature. In order to compare our work with those described in the literature, it was studied the sensitivity of the parameters of the various models. The continuation of the study assumed identification of the unknown parameters of the model and judicious selection of the stress-strain curve selected in order to avoid the problems of identification. Therefore the division of the model into sub models is strategic, because it makes possible to identify per pieces the vector parameter of the total model. The last part of the study aims at verification of the correctness of model proposed. For that, various types of yarns were tested according to their origin: natural, or man-made.

INTRODUCTION

The leading part of the industry textile/clothing is steered on the manufacture of clothes and by the same one of fabrics. For a long time textile engineers have known that the properties of fabrics have an essential influence on the manufacturing process and as well on the appearance of ready-to-wear clothing. Having the knowledge of textiles, they try to predict the behavior of clothes. Although the modeling of materials is not a particularly recent subject (Tchebichef 1878; Lucas 1878), the level of knowledge is still too limited. The structure of fabrics and properties are very complex, i.e. :

- the variety of the modes of crossing of the yarns under the shape of different weaves (plain, twill, satin),
- the variety of the used materials, that are natural as cotton, wool, linen, silk, and synthetic materials as the PA, PES, PP,
- the physical characteristics of the materials, which compose the fabric.

The economic and industrial reasons cause a strong motivation for research on the models of yarn and fabrics, the objective of which is to understand the dynamic behavior of

these last ones. For that purpose, it is necessary to move the real world closer to the virtual world by optimizing the error of interpretation between these two worlds. To characterize the properties of fabrics, first it is necessary to take into account the structure of these ones. In that case, its geometry should be established with a big precision basing on the considerations that a fabric is a result of the intertwining of a warp and a weft yarns. The step of modeling of the fabric takes into account these remarks, but requires the modeling of the yarn.

THEORETICAL APPROACH

The analysis of the stress-strain curves allows us to established our model of stress-strain curve. In order to have an approach system, we observed the stress-strain curve like a signal according to time, i.e., the constraint is compared with the dynamic response of a system, the deformation corresponds to the time. In a first phase, we analyzed the models established by the others researchers. We studied the sensitivity of the parameters of each model according to its curve of representation. This approach makes it possible to estimate the influence of each parameter in the time and to check, if our initial assumptions to divide time space into three distinct zones are perceptible in the other models through the parameters. This study will guide us to different submodels intervening in each zone from our total model. To facilitate the comparison of each model with our model, the whole simulation programs was written using a software MATLAB. The aim of the study is to show the importance of each variable, i.e., its their influence on the shape of the curve.

From many practical tests on various samples, we could estimate a general shape of the stress-strain curve of the yarn presented of Figure 1. The shape of this curve is divided into three distinct parts, which we describe bellow.

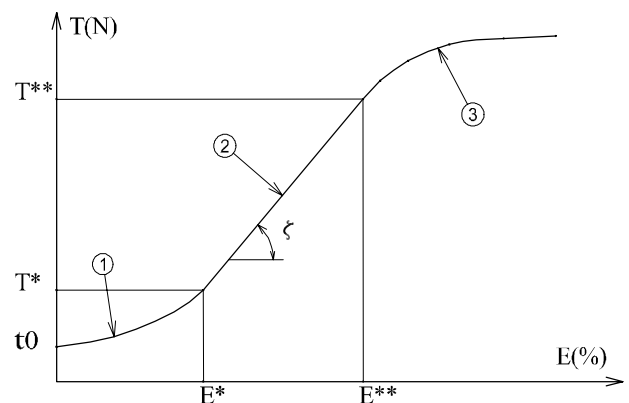


Figure 1 : Model of the stress-strain curve

The first zone

The first zone corresponds to the moment of starting of the tensile tests of yarn. The yarn is a fiber assembly. Fiber arrangement in the yarn is partially ordered, although during the phase of design of the yarn, the process of spinning tries to preserve a regular torsion, which makes it possible to have a homogeneity of this one. During the tensile tests, the fibers are subjected to longitudinal constraints. These constraints result in an alignment of the fibers, which generate the interaction of fibers as well as phenomena of slippage. This zone corresponds to the "pre-tensioning" of fibers, it is acted in a fact of their orientation according to the axis of deformation. Thus the fibrous structure tends towards a new space orientation (energy minimum), before the clean deformation of the yarn starting from the point E *.

The second zone

This zone explains the elastic character of the yarn (zone with linear elasticity); the relationship between a constraint and deformation is then proportional. The proportionality factor is represented by the Young modulus. The value of this modulus is constant until a certain maximum limit, beyond which the deformation is not elastic any more (plastic zone). The passage between the elastic zone and plastic deformations zone is identified by the point E **.

The third zone

The last zone of the stress-strain curve reveals nonlinear phenomena, which are explained by the damage of fibers. During these phases, there is a progressive destruction of fibers starting from the weakest one.

TRACTION MODELS

Vangheluwe's model

Figure 2 represents the model of Vangheluwe (Vangheluwe 1992; Vangheluwe 1993).

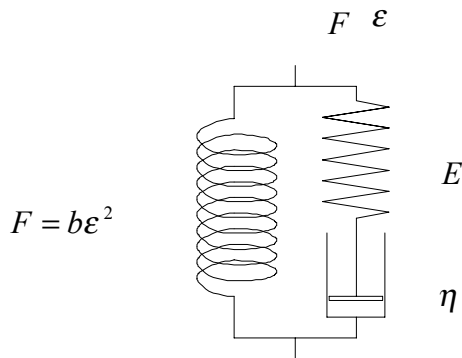


Figure 2 : Vangheluwe's model

This model is a sequence of various models. It is composed of the Hooke's model (elastic model) connected in series with a viscoelastic model (Maxwell's model), in parallel with term F grafted on the unit in order to take into account the non linearities due to stretching forces. The equation of this model is :

$$F(\epsilon) = F_0 + \eta r \left(1 - \exp\left(-\frac{E\epsilon}{\eta}\right) \right) + b\epsilon^2 \quad (1)$$

We introduced some variables :

$$\eta r = A$$

$$-\frac{E}{\eta} = -B \quad (2)$$

$$b = C$$

The final equation becomes :

$$P(\epsilon) = P_0 + A[1 - \exp(-B\epsilon)] + C\epsilon^2 \quad (3)$$

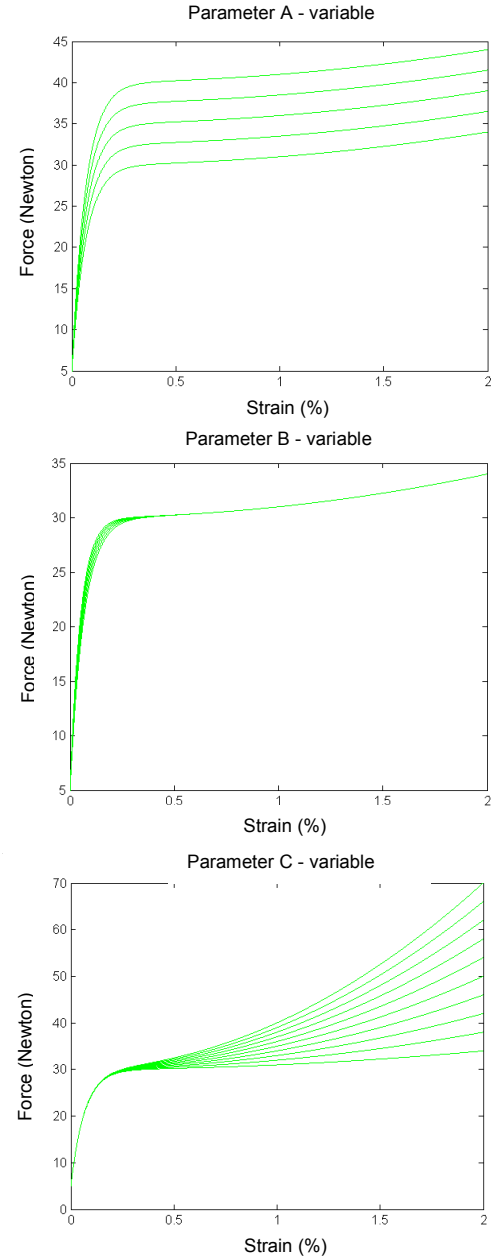


Figure 3 : Curve of sensitivity of the Vangheluwe's model

Sensitivity of Vangheluwe's model

In the Vangheluwe's equation, it appears the expression $A(1 - \exp(-B\epsilon)) + C\epsilon^2$, which is very close to the Maxwell's model supplemented by a nonlinear spring. The factor P_0 describes the initial state of the pre-tensioning of

fibers in the yarn. From the analysis of curve sensitivity (Figure 3), we could get the following conclusions :

- The effect of variable A can be assimilate to the static gain of the system,
- The effect of the variable B can be equivalent to the time constant of the system,
- The effect of the variable C is strongly related to the nonlinear phenomenon translated by the expression $C\varepsilon^2$. C operates on the final part of the curve and reacts in a contradictory way with of variable A; while being opposed to this one at the end of the curve. It gives again a new dynamics in the system.

While comparing with our model, we note that :

- The variable B acts on the first and second zone of stress-strain curve,
- The variables A and C intervene in the third zone.

It is important to note that the actions on the quoted zones are independent. The factor P_0 defines the initial state of the test and corresponds to the setting in progressive tension of fibers in the yarn. The Vangheluwe's model appears incomplete, because it does not distinguish zone 1 and 2 from the stress-strain curve, i.e., the zone of fiber alignment and the zone of elastic linear strain.

Zurek's model

The Zurek's model (Zurek and Aksan 1975) is a relatively complete rheological model (Figure 4), because it integrates the friction and inertia effects by the terms (T) and (M); and put in parallel with the Kelvin Voigt's model (η, K_2), supplemented in series by an elasticity (K_1) assimilating the Hookean region.

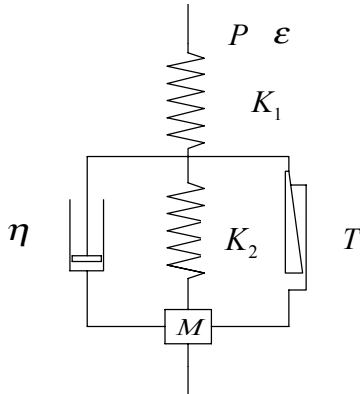


Figure 4 : Zurek's model

Particular solution of the model :

$$P(\varepsilon) = A\varepsilon + B(C\varepsilon - B)\exp(-a\varepsilon) \quad (4)$$

Sensitivity of the Zurek's model

Analysis of the curves of Figure 5 showed that :

- The variation of A is similar with that of the parameter C of Vangheluwe's model with more marked linearity,
- The variation of B is regarded as an action on the static state of the system (static gain),
- The variation of the variable C expresses the first overtaking of the signal, which one finds on a system of the second order,

- The variation of a is associated with the time constant of the system.

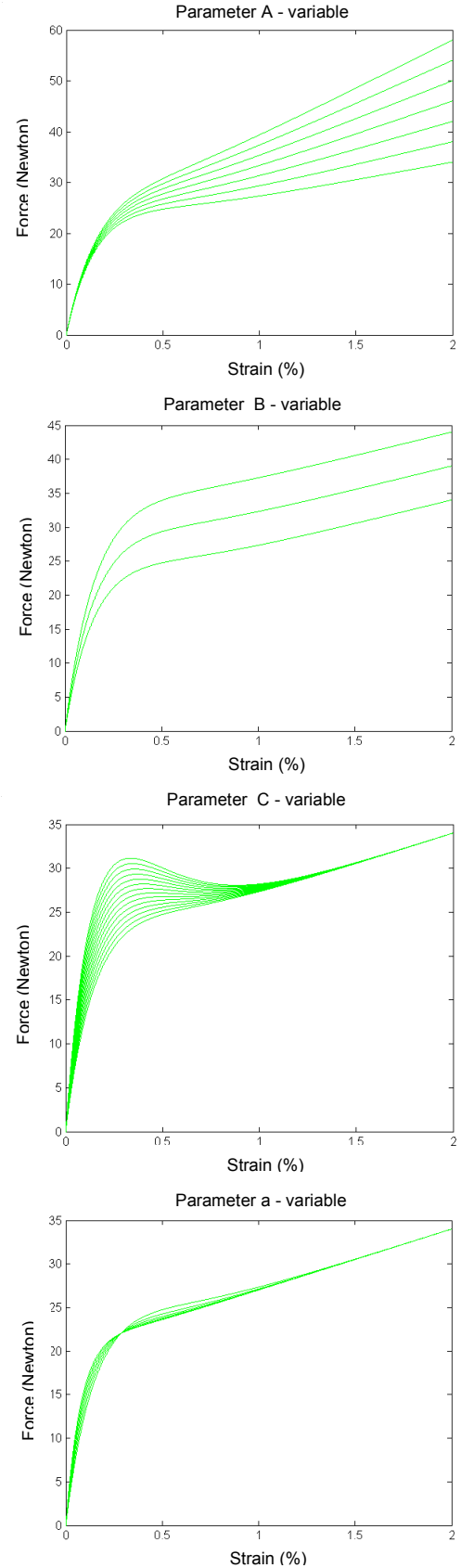


Figure 5 : Curve of sensitivity of the Zurek's model

While comparing with our model, except the parameters from the same point of view as the Vangheluwe's model, only the parameter C brings a particular dynamics at the linear end of

the zone (between zone 2 and 3). This effect is noted only on some yarns.

Manich models

Manich (Manich 1999) differently translated the nonlinearities by different interpretation of the Vangheluwe's and Zurek's models in order to integrate the phenomenon into fibers in the yarn. It then proposes two modified models :

Vangheluwe's model modified by Manich

We treated this model in a similar way. The difference between the basic model and the following modified model :

$$P(\varepsilon) = P_0 + A[-\exp(-B\varepsilon^D)] + C\varepsilon^E \quad (5)$$

appears only on the exponent of the terms of deformation. Also, we endeavored to study only the complementary variables D and E.

Sensitivity of the Vangheluwe's model modified by Manich

Curves of Figure 6 show us how these two terms influence the curve.

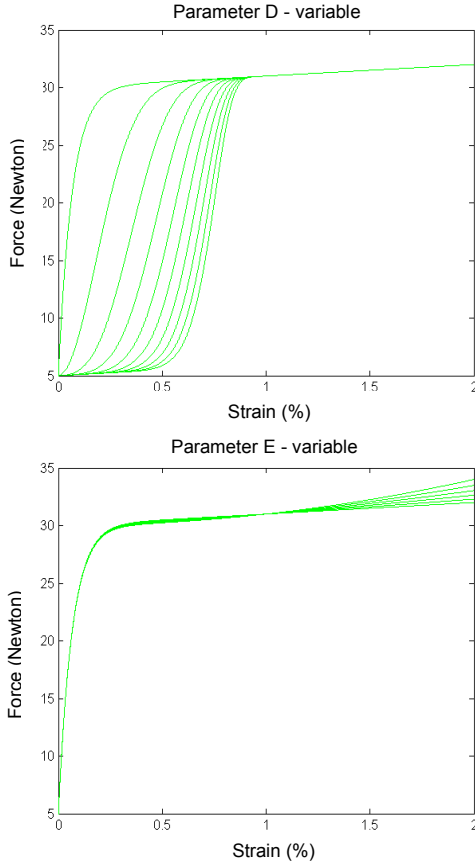


Figure 6 : Curve of sensitivity of the Vangheluwe's model modified by Manich

The conclusions, which we get from the modifications are as follows :

- D plays an important role at the beginning of the curve. It introduces into a curve a horizontal tangency at the beginning of curve, what enables to manage in a gentle way the time constant and to introduce a late time,

- E acts in the nonlinear part by giving again a more constant dynamics at the end of the signal.

While comparing with our model, we note that :

- The variable D creates two zones 1 and 2, placed in one zone of the basic model,
- E accentuates the nonlinear effects only in zone 3.

Zurek's model modified by Manich

Modification made to the model is :

$$P(\varepsilon) = A\varepsilon + B + (C\varepsilon - B)\exp(-a\varepsilon^D) \quad (6)$$

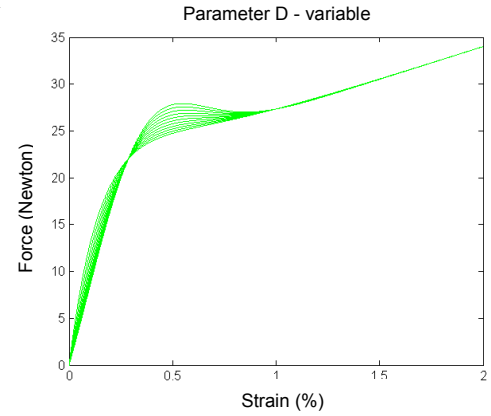


Figure 7 : Curve of sensitivity of the Zurek's model modified by Manich

Sensitivity of the model

The variation of D (Figure 7) amplifies the effects of the parameter C of the basic model, but in a way shifted in time, while acting on the transitory state by parameter a. This parameter combines judiciously the effects of the parameters C and a of basic model. Like it was noticed at the time of study of the modified Vangheluwe model, the parameter D makes also possible to reveal two zones 1 and 2 of our model, but in manner less pronounced at the beginning of curve.

Legrand's model

It proposes a system of equations, in which each equation gives a representation of the phases previously described (Legrand 2000). Its approach is based on a study of a shape of the practical curves. The Legrand's model is written as follows :

$$f_v(\xi, \xi, t) = \begin{cases} a\xi + b + c \sin(d(\xi - e)) \exp^{f(\xi - e)} + A \exp^{Bt} + C \exp^{Dt}, & \text{si } \xi \geq 0 \\ F(\xi) = \exp^{\alpha(\xi - \beta)} - 1, & \text{si } \xi < 0 \end{cases} \quad (7)$$

avec, $\alpha = \gamma \ln(\xi) + 1$

in which appears independently :
the viscoelasticity term

$$a\xi + b + c \sin(d(\xi - e)) \exp^{f(\xi - e)} \quad (8)$$

the relaxation terms

$$A \exp^{Bt} + C \exp^{Dt}, \text{ si } \xi \geq 0 \quad (9)$$

as well as the terms translating the hysteresis of the yarn

$$F(\xi) = \exp^{a(\xi-\beta)} - 1, si \xi \leq 0 \quad (10)$$

Sensitivity of the Legrand's model

The sensitivity of each parameter is presented in Figure 8.

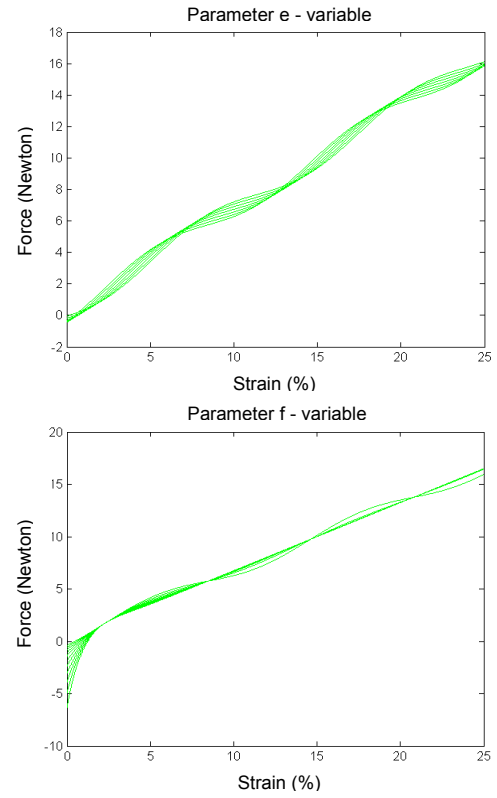
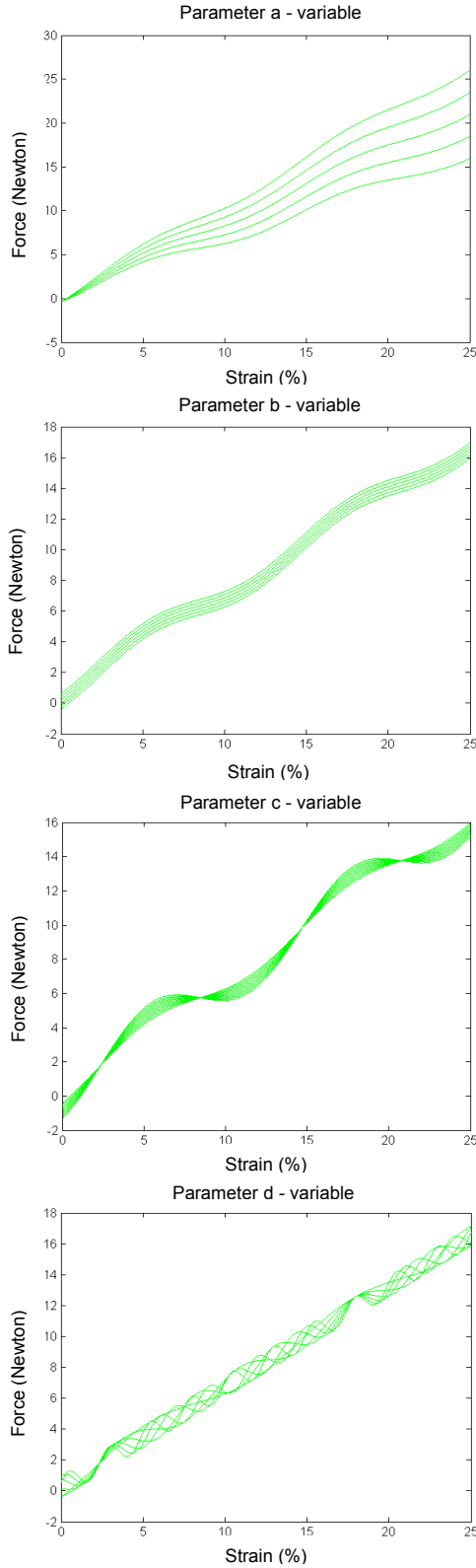


Figure 8 : Curve of sensitivity of the Legrand's model

The parametric analysis brings us to the following conclusions :

- a is a parameter, which makes it possible to manage the rising time of the signal. It adapts the slope of the continuous component (level, component),
- b position by an offset the sinusoidal component compared to the continuous component,
- c manages the amplitude of the sinusoidal component,
- d defines the period of the sinusoidal component,
- e represents the term of dephasing of the sinusoidal component,
- f is the damping ratio of the sinusoidal component.

PROPOSED MODEL

The shape of curve given by figure 1 is described by the following equations :

$$\left\{ \begin{array}{l} Y = \zeta \left[\left(1 - \frac{T^*}{\zeta E^*} \right) E + \frac{2T^*}{\zeta} - E^* \right] \left(\frac{E}{E^*} \right) + T_0 \\ \quad \quad \quad \text{if } E \leq E^* \\ Y = \zeta (E - E^*) + T^* \\ \quad \quad \quad \text{if } E^* \leq E \leq E^{**} \\ Y = \zeta \left[1 - \left(a + b \exp \left(-\frac{E}{c} \right) \right) \right] * (E - E^{**}) + T^{**} \\ \quad \quad \quad \text{if } E \geq E^{**} \end{array} \right. \quad (11)$$

Sensitivity of the Model proposed

Our model integrates the various dynamic components present in the other models. Only, a study of the sensitivity of

the parameters of each model enables to show them (Figure 9).

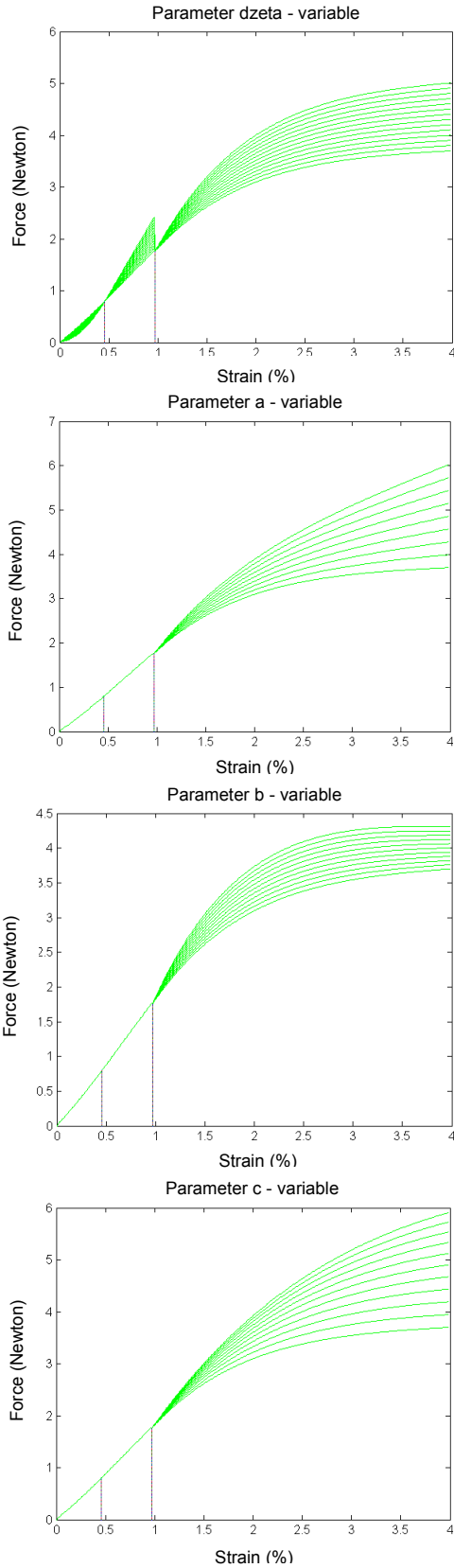


Figure 9 : Curves of sensitivity of the model proposed

So, we are going to attempt to show that the parameters of our model contribute to a good representation of the curve by acting in a similar way in the parameters of the other models.

- The parameter ζ is a variable, which manages the slope of the identical curve in the 3 zones. That enables to keep a continuity of the function of representation at the discontinuity moments of submodels, the tangents at the points of passage must be the same ones.
- The interest to act on the parameters T^* and T^{**} is to avoid the break at the moments of discontinuity.

Parameters a, b, c act only in the third part of curve. This part in nonlinear matter (à caractère non linéaire) is managed in the following way, as show in Figure 8.

- The parameter b as a time constant influence the behavior of the signal and balance the final level of the curve, i.e., the equivalent of a variable static gain.
- Parameter a tends to give a non null slope of the signal, whereas the parameter c deadened this effect especially at the end of the signal.

The combination of these three parameters manages the shape of the curve.

PROCEDUREE AND IDENTIFICATION ALGORITHM

The identification of the parameters of our model was carried out using the MATLAB software. The procedure of identification to be followed is described in Figure 10. The model of the stress-strain curve was judiciously selected in order to avoid the problems of identification.

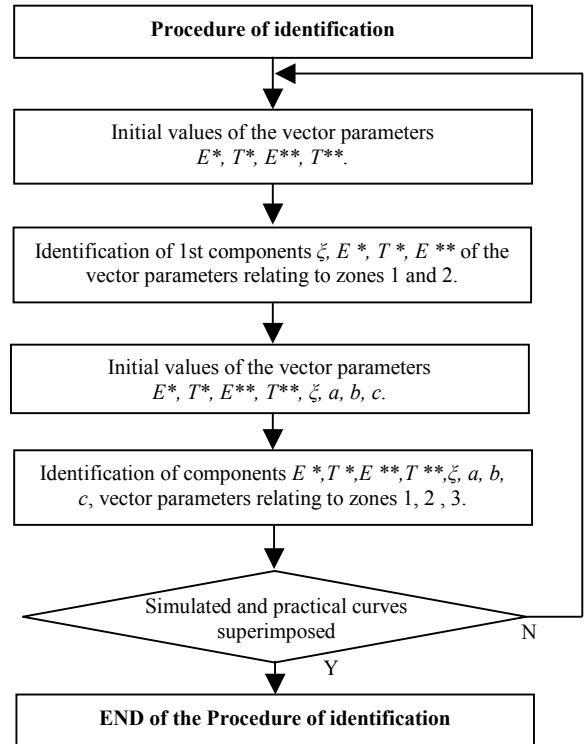


Figure 10 : Procedure of identification

For that, the division of the model in submodels is a strategy, because it enables an piecewise identification of the vector parameter of the global model. In the case of a single model integrating a great number of parameters, as was shown in bibliography, complex methods of identification would be an implementation to avoid the parametric compensations due to the big number of parameters (Bruniaux 1988).

In our case, the identification is carried out in several steps. The first consists of identifying the parameters of submodels relating to zones 1 and 2 of the stress-strain curve. The second corresponds to the identification of the parameters of the model penny relating to zone 3. The flow chart of Figure 10 recalls the methodology described by the various points :

- Analysis of a stress-strain curve in order to locate the linear zone (2) and to obtain the approximate values of the parameters E^* , T^* , E^{**} , T^{**} ,
- Introduction of the initial vector parameters necessary to start the program of identification. Only, submodels 1 and 2 are used,
- Identification of the parameters E^* , T^* , E^{**} starting from zone 1 and 2 of the stress-strain curve (method of least squares nonlinear). The results correspond to a first approximation of these data. The interest of this step aims at facilitation of the convergence of the identification algorithm,
- Introduction of the initial vector parameters necessary to the identification of sub-models 1,2 and 3. The results from the preceding identification are reinjected in this vector,
- Identification of the parameters E^* , T^* , E^{**} , T^{**} , a , b , c , by accounting all points of the stress-strain curve (zones 1-2-3). We always use the nonlinear least squares method,
- Presentation of the values obtained by the superposition of the real and simulated curves,
- Acceptance or not of the results. If not, it is necessary to take the procedure again as in the point 1.

THEORICAL AND SIMULATED RESULTS

In the following section, we validate our model with various types of yarn (PES, COTTON and TENCEL) under various configurations. Work of the authors previously quoted shows that the practical (red) and simulated (blue) stress-strain curves can have various morphologies. Also, we wished to test the influence of the origin of the yarn in order to check the impact of its parameters on the curve shape. The curves present the changes evolution of the value of the vector parameters :

- before the identification, initial vector parameters,
- during the identification, results from phase 1 of the identification (zone 1 and 2),
- after the identification, results from phase 2 of the identification (zone 1, 2 and 3).

In a clothing application, it is possible to classify yarns according to the fiber origin. Three categories appear : natural, artificial, synthetic.

The origin of the yarn : natural - COTTON

Initial vector parameter :

$\xi=1.746$, $T^*=0.6038$, $E^*=0.3323$, $E^{**}=0.84$, $T^{**}=0.76$, $a=0.1388$, $b=1.22$, $c=3$

Intermediate result :

$\xi=1.9211$, $T^*=0.7986$, $E^*=0.4557$, $E^{**}=1.0000$

Final result :

$\xi=1.9247$, $T^*=0.8062$, $E^*=0.4600$, $E^{**}=0.9700$, $T^{**}=1.7607$, $a=0.1718$, $b=1.5703$, $c=1.7612$

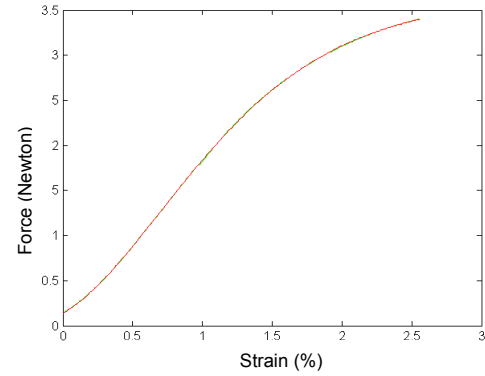


Figure 11 : Cotton stress-strain curves

The origin of the yarn : artificial – TENCEL

Initial vector parameter :

$\xi=1.746$, $T^*=0.4$, $E^*=0.2$, $E^{**}=0.8$, $T^{**}=1.4$, $a=0.1388$, $b=1.22$, $c=3$

Intermediate result :

$\xi=1.7032$, $T^*=0.3913$, $E^*=0.1474$, $E^{**}=0.8000$

Final result :

$\xi=1.7187$, $T^*=0.2230$, $E^*=0.0510$, $E^{**}=0.7696$, $T^{**}=1.4187$, $a=-0.0368$, $b=1.0494$, $c=2.8014$

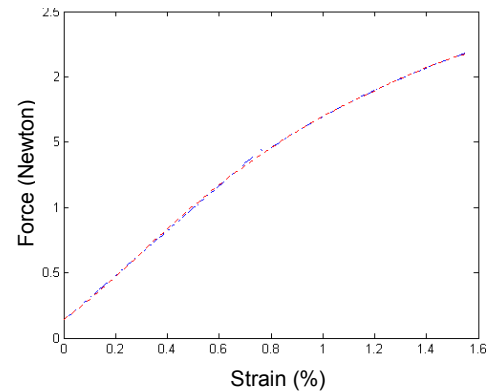


Figure 12 : Tencel stress-strain curves

The origin of the yarn : synthetic – PES

Initial vector parameter :

$\xi=1.746$, $T^*=1.3$, $E^*=2.5$, $E^{**}=3.7$, $T^{**}=1.6$, $a=0.1388$, $b=1.22$, $c=3$

Intermediate result :

$\xi=0.2342$, $T^*=1.1863$, $E^*=2.0708$, $E^{**}=3.7000$

Final result :

$\xi=0.2334$, $T^*=1.1874$, $E^*=2.0745$, $E^{**}=3.8602$, $T^{**}=1.6038$, $a=2.1467$, $b=-3.3471$, $c=3.898$

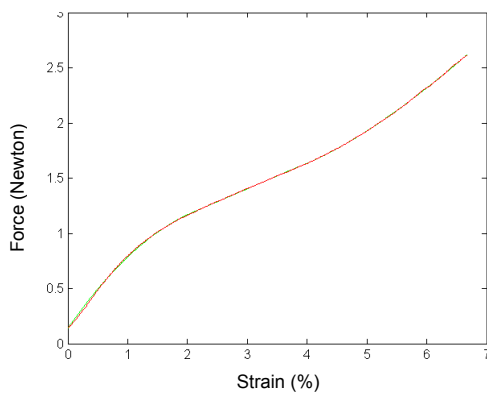


Figure 13 : PES stress-strain curves

The superposition of the curves shows that the model is adapted perfectly in almost every type of used yarn. Only the average results of each yarn are presented. The following table summarizes the value of the parameters for the PES, Tencel and red Cotton.

Yarn	ζ	T*	E*	E**	T**	a	b	c
1	2	3	4	5	6	7	8	9
PES	0.215	1.742	4.899	6.015	1.989	2.810	-6.460	4.044
	0.233	1.187	2.074	3.860	1.603	2.146	-3.347	3.898
	0.282	0.851	1.359	5.895	2.207	1.144	0.219	0.086
	0.315	0.874	1.131	2.967	1.432	1.706	0.219	0.086
	0.274	0.909	1.509	3.111	1.288	1.178	0.219	0.086
	1.703	0.771	0.368	0.761	1.405	0.315	0.872	1.288
TENCEL	1.285	1.265	0.739	0.800	1.350	0.509	0.587	1.311
	2.087	0.507	0.184	0.670	1.486	0.174	1.229	1.149
	2.022	0.393	0.137	0.671	1.448	0.376	0.726	1.338
	1.960	1.456	0.645	0.668	1.470	0.327	0.760	1.378
	1.746	0.597	0.328	0.987	1.722	0.176	1.381	1.645
	1.921	0.806	0.459	0.981	1.788	0.091	1.540	2.049
COTTON red	1.705	0.583	0.302	0.990	1.737	0.179	1.469	1.507
	1.662	0.750	0.463	1.060	1.722	0.182	1.637	1.446
	1.924	0.806	0.460	0.970	1.760	0.171	1.570	1.761

CONCLUSION

We propose a model for the yarn stress-strain curve. We tried to integrate the textile character into it by interpreting this curve in the form of various models. This differentiation is due to a change of yarn state at the stretching time. So the model is distinguished very clearly from the other models exposed in the literature. Also, in order to compare our work with those described in the bibliography, it was necessary to study the sensitivity of the parameters of the various models. The continuation of the study went on the identification of the unknown parameters of the model. We set up a strategy of development in order to avoid the effects of parametric compensation very often observed when the number of parameters of a model becomes consequent. The division of the model into submodels contributes to the correct operation of the identification, because we use a method of identification per pieces with submodels having the small number of parameters. We think that the models presented in the bibliography can lead to erroneous results seen the significant number of parameters and taking into consideration their curve of sensitivity. The last part of the study aims to check the correct operation of the model. For this purpose, various types of yarns were tested: natural, artificial, or synthetic. The curves presented show that the model was adapted successfully for almost all of the tested

yarns. This part was supplemented by a study of the influence of the twist on the stress-strain curve. The results show that the effects of these factors should be taken into account in the future work. They act in quite precise zones of our model on precise parameters. Also, it would be interesting to find a correlation between the influenced parameters and the factors acting on them.

REFERENCES

- E. Lucas, Sur la géométrie du tissu, 1878.
- Tchebichef, Sur la coupe des vêtements, 1878,
- L.Vangheluwe, Study of the Time Dependent Mechanical Properties of yarns for weaving, *Doctoral thesis*, 1992, University Ghent.
- L.Vangheluwe, Relaxation and inverse relaxation of yarn after dynamic loading, *Textile Research Journal*, 63(9), 1993, 552-556.
- W.Zurek & S.Aksan, A rheological model of viscose rayon, *Journal Appl. Polymer Science*, 19, 1975, 3129-3137.
- A.Manich, Viscoelastic behavior of polypropylene fibres, *Textile Research Journal*, 69(5), 1999, 325-330.
- X.Legrand, Contribution à la modélisation dynamique et à l'identification de tissus techniques, *Thèse de l'université*, 2000 Université des Sciences et Technologies de Lille.
- P.Bruniaux, Contribution à la modélisation non linéaire à l'identification et à la commande en boucle ouverte de moteur pas à pas, *Thèse de docteur de 3^{ème} cycle*, 1988, Université des Sciences et Technologies de Lille.

BIOGRAPHY

JEAN MARIE CASTELAIN received his engineer degree from the Higher National School of Arts and Trades (ENSAM) in 1976 and his Ph.D. degree in Mechanical Engineering from University of Lille (USTL) in 1981. Since 1987, he is Doctor Ability and University Professor. Currently, he is Head of Higher National School of Works and Textile Industries (ENSAIT) of Roubaix (France), since 1997.

PASCAL BRUNIAUX was born in 1959 and he received his Ph.D. degree in Automatic and Industrial Computing from University of Lille (USTL) in 1985. Currently, he is Assistant Professor in Dressing of Higher National School of Works and Textile Industries (ENSAIT) of Roubaix (France) since 1989. His research fields of interest are Modeling of yarn, Modeling of fabric, Nonlinear identification, Virtual reality clothing.

LATE PAPERS

SIMULTANEOUS PERTURBATION STOCHASTIC APPROXIMATION FOR REAL-TIME OPTIMIZATION OF MODEL PREDICTIVE CONTROL

Irina Baltcheva
Felisa J. Vázquez-Abad¹ (member of GERAD)
Université de Montréal
DIRO, CP 6128 succ Centre Ville,
Montreal, QC, H3C 3J7 Canada
{baltchei,vazquez}@iro.umontreal.ca

Smaranda Cristea
César De Prada
Universidad de Valladolid
DISA, P Prado de la Magdalena s/n,
Valladolid, 47005 Spain
{smaranda,prada}@autom.uva.es

ABSTRACT

The aim of this paper is to suggest a global optimization method applied to a model predictive control problem. We are interested in the Van der Vusse reaction found in multiple chemical processes. The control variables (temperatures and input flow rates) are real time continuous processes and a target reference level must be reached within certain operational constraints. The canonical model discretises time using a sampling interval, thus translating the control problem into a non-linear optimization problem under constraints. Because of the non-linearity of the cost function, common methods for constrained optimization have been observed to fail in experiments using ECOSIM to simulate the production process. The controllers do not achieve their optimal values and the numerical optimization based on approximating gradients and Hessians cannot be performed in real time for the operation of the plant to be successful. In this research we implement a methodology for global optimization adding noise to the observations of the gradients, which will perform much better than deterministic methods.

INTRODUCTION

Model Predictive Control (MPC) is now recognized in the industrial world as a proven technology, capable of dealing with a wide range of multivariable constrained control problems. Nevertheless, most of the industrial controllers are based on linear internal models which limits its applicability. Because of it, non-linear model predictive control (NMPC) has received a lot of attention in the latest years, both from the point of view of its properties [2] and implementation. Referring to this last aspect, the main drawback is the computational burden that NMPC implies. While linear MPC with constraints can solve the associated optimization problem each sampling time using QP or LP algorithms for which very efficient codes are available, NMPC relies on non-linear

programming (NLP) methods such as SQP, that are known to be far more CPU demanding. Several schemes have been proposed to deal with this problem, among them the well known sequential and simultaneous approaches.¹

For sequential solutions, the model is solved by integration at each iteration of the optimization routine. Only the control parameters remain as degrees of freedom in the NLP. Simulation and optimization calculations are performed sequentially, one after the other. The approach can easily be coupled with advanced simulation tools. In contrast, simultaneous model solution and optimization includes both the model states and controls as decision variables and the model equations are appended to the optimization problem as equality constraints. This can greatly increase the size of the optimization problem, leading to a trade-off between the two approaches. In both cases, computation time remains a difficulty in order to implement NMPC in real processes.

This paper shows a global optimization method oriented to reduce the difficulties associated with the computation of the gradients, in order to facilitate the implementation of NMPC algorithm, using the sequential approach, applied to a benchmark problem: Van der Vusse reactor.

MODEL DESCRIPTION

The Van der Vusse reaction is described in detail in [4] and the references therein. To summarize, there is a substance A in input, a chemical reaction and a substance B in output.

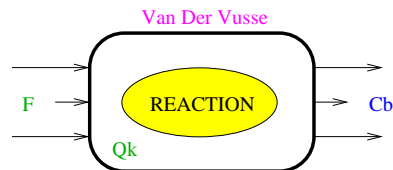


Figure 1: Van der Vusse

¹Also Principle Investigator, Department of Electrical and Electronic Engineering, The University of Melbourne.

We'll denote by:

c_B the concentration of product B (controlled)

c_A the concentration of product A (measured)

T the temperature in the reactor (measured)

T_K the temperature in the coolant (measured)

F the input flow of product A (manipulated)

Q_K the heat removal (manipulated)

$c_{A0}, T_0, k_1, k_2, k_3, k_w, \rho, c_p, V_R, \Delta H_1, \Delta H_2, \Delta H_3, m_K$ constants.

To the input we associate the **control variables** F and Q_K . The **controlled variable** c_B is associated with the output. Thereby, we want to control the output c_B by manipulating the input F and Q_K .

To describe the evolution of this dynamical system, we introduce the non-linear differential equations related to mass and energy conservation:

$$\begin{aligned} c_A' &= \frac{F}{V_R}(c_{A0} - c_A) - k_1 c_A - k_3 c_A^2 \\ c_B' &= -\frac{F}{V_R} c_B + k_1 c_A - k_2 c_B \\ T' &= \frac{F}{V_R}(T_0 - T) - \frac{1}{\rho C_p}(k_1 c_A \Delta H_1 + k_2 c_B \Delta H_2 + \\ &\quad + k_3 c_A^2 \Delta H_3) + \frac{k_w A_R}{\rho C_p V_R}(T_K - T) \\ T_K' &= \frac{1}{m_K C_{pK}}(Q_K + k_w A_R(T - T_K)) \end{aligned}$$

The concentration of product B must not exceed some upper and lower limits. This gives the following constraints, which will be included in the objective function later on:

$$l \leq c_B \leq L.$$

MODEL BASED PREDICTIVE CONTROL

Let us denote by:

$x_t = (c_A, c_B, T, T_K)$ the state at time t

$y_t = (c_B)$ the controlled variable at time t

$u_t = (F, Q_K)$ the control at time t (manipulated variable)

r : reference level.

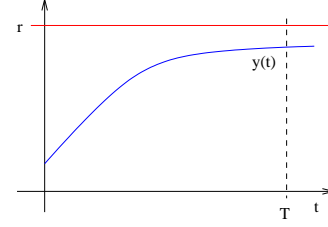


Figure 2: Control

The objective of the predictive control is to find the future optimal control sequence $u_t^* = (u^*(i), i = 0, 1, \dots, N(T))$ over a finite horizon time T , which minimizes the *quadratic error* between the controlled variable y_t and its reference level r . In addition, u_t^* must assure that the trajectory of y_t is smooth, i.e. the range of values taken by the manipulated variables F and Q_K varies gradually and does not go from very high values to very low ones. The objective function is then:

$$J(x_t, y_t, u_t) = \int_0^T \|y_t - r\|^2 dt + \beta \int_0^T \|\dot{u}_t\|^2 dt,$$

where $\dot{u} = \frac{du}{dt}$, T is the prediction horizon and $\beta \geq 0$ is constant. The optimization is done over the set of feasible controls \mathcal{U} :

$$\min_{u \in \mathcal{U}} J(x_t, y_t, u_t)$$

The *receding horizon* mode of operation is used here: once the optimal control sequence $u_t^* = (u^*(i), i = 0, 1, \dots, N(T))$ is found, only $u_t^*(0)$ is applied to the system at time t . At time $t + 1$, another optimal sequence is found and again, only the first component is applied. The control u_t is kept constant during a so called *sampling time* h , when all the simulation parameters are kept at a fixed value. In the experiments already done, the sampling time was 20 seconds. At the next sampling time $t + h$, another optimal sequence is found and again, only the first component $u_t^*(0)$ is applied. Thus, the control is better adapted to the system's actual state and the noise is reduced. Notice that the time horizon is moved ahead (receding-horizon): $T(t + h) = T + h$.

In order to solve the problem it is necessary to parameterize the manipulated variable u_t , otherwise an infinite number of decision variables would appear in the problem. An usual approach is to discretize u along the control horizon N_u when the input remains constant over the sampling period h :

$$\begin{aligned} u_t &= u(k), \quad kt \leq t < (k+1)t, \\ u(k) &= u(N_u - 1), \quad \forall k > N_u - 1. \end{aligned}$$

Thus, the constrained optimization problem, subject to the continuous model equations and to the typical restrictions

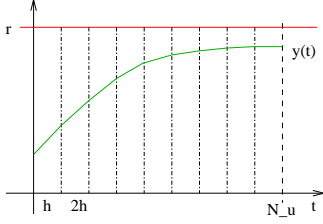


Figure 3: Discretized Control

applied to the manipulated and controlled variables, can be written as:

$$\min_{u(\frac{k}{k}), \dots, u(\frac{k+N_u-1}{k})} J = \int_{t_k}^{t_k+N\tau} \gamma [y(t) - r(t)]^2 dt + \sum_{i=0}^{N_u-1} \beta [\Delta u(k+i|t)]^2$$

$$\begin{aligned} y_{min} &\leq y(k) \leq y_{max} \\ u_{min} &\leq u(k) \leq u_{max} \\ \Delta u_{min} &\leq \Delta u(k) \leq \Delta u_{max} \end{aligned}$$

After penalizing the constraints on the controlled variable y , the objective function becomes:

$$\begin{aligned} \min_u J &= \int_{t_k}^{t_k+N\tau} \gamma [y(t) - r(t)]^2 dt \\ &+ \sum_{i=0}^{N_u-1} \beta [\Delta u(k+i|t)]^2 \\ &+ \sum_{j=0}^{N_u-1} \alpha_1 [y_{max} - y(k+j|t)]^2 \\ &+ \sum_{l=0}^{N_u-1} \alpha_2 [y_{min} - y(k+l|t)]^2 \end{aligned}$$

where β and γ are non-negative constants, and α_1, α_2 represent a penalty function. Remark that this optimization problem cannot be solved analytically. In fact, it is hard to calculate the gradient $\nabla_u J$, because y is non-linear in u . This is why we need to estimate it.

In order to treat the control constraints, we'll make a projection of the Lagrangian function J over the set of feasible controls. We'll truncate it in the sense that if $u(k) \geq u_{max}$, we'll set $u(k) = u_{max}$, and if $u(k) \leq u_{min}$, we'll set $u(k) = u_{min}$.

Previous approaches to the optimization problem were based on the SQP algorithm implemented in the NAG library, which uses finite differences to estimate the gradient ∇J . This approach has the disadvantage of losing precision and is quite slow because of the large number of func-

tion evaluations. Also, it may get trapped in local minima.

SYSTEM REQUIREMENTS

To see how fast the optimization must be, let us summarize the simulation by the algorithm below:

WHILE (simulation time) DO

WHEN (sampling) DO

1. Calculate the optimal control ($\min\{u \in \mathcal{U} : J(x_t, y_t, u_t)\}$);
2. Apply the control u_t^* ;
3. Measure C_a, T, T_K ; $x_t \leftarrow (C_a, T, T_K)$;
4. Take new value of C_b ; $y_{t+1} \leftarrow C_b$;
5. $t = t + h$; go to 1.

END WHEN

END WHILE

It is clear that the optimal control must be found in less than $h = 20$ seconds (the sampling time) if we want the control to be done online. The optimization must be fast and precise.

The Van der Vusse model presents important nonlinearities which makes the problem quite difficult. Another source of problems is the possible existence of local minima. Among others, this is a reason why we'll apply a method of global optimization. In the next section, we present this method, which is known for its efficiency in high dimensional problems. It could replace successfully the finite differences approximations in the Van der Vusse model and will perform as well in the presence of a larger number of variables.

SIMULTANEOUS PERTURBATION STOCHASTIC APPROXIMATION (SPSA)

SPSA is a descent method capable of finding global minima. Its main feature is the gradient approximation that requires only two measurements of the objective function, regardless of the dimension of the optimization problem. Recall that we want to find the optimal control u^* , with loss function $J(u)$:

$$u^* = \operatorname{argmin}\{J(u) : u \in \mathcal{U}\}$$

Both Finite Differences Stochastic Approximation (FDSA) and SPSA use the same iterative process:

$$u_{n+1} = u_n - a_n \hat{g}_n(u_n),$$

where $u_n = ((u_n)_1, (u_n)_2, \dots, (u_n)_p)^T$ represents the n^{th} iterate, $\hat{g}_n(u_n)$ is the estimate of the gradient of the objective function $g(u) = \frac{\partial}{\partial u} J(u)$ evaluated at u_n , and $\{a_n\}$ is a positive number sequence converging to 0. If u_n is a p -dimension

vector, the i^{th} component of the symmetric finite difference gradient estimator is:

$$\text{FD: } (\hat{g}_n(u_n))_i = \frac{J(u_n + c_n e_i) - J(u_n - c_n e_i)}{2c_n},$$

$1 \leq i \leq p$, where e_i is the unit vector with a 1 in the i^{th} place, and c_n is a small positive number that decreases with n . With this method, $2p$ evaluations of J for each g_n are needed. Clearly, when p is large, this estimator loses efficiency.

The simultaneous perturbation estimator uses a p -dimensional random perturbation vector $\Delta_n = ((\Delta_n)_1, (\Delta_n)_2, \dots, (\Delta_n)_p)^T$ and the i^{th} component of the gradient estimator is, $1 \leq i \leq p$:

$$\text{SP: } (\hat{g}_n(u_n))_i = \frac{J(u_n + c_n \Delta_n) - J(u_n - c_n \Delta_n)}{2c_n (\Delta_n)_i}.$$

Remark that FD perturbs only one direction at the time, while the SP estimator disturbs all directions in the same time (the numerator is identical in all p components). The number of loss function measurements needed in the SPSA method for each g_n is always 2, independent of the dimension p . Thus, SPSA uses p times fewer function evaluations than FDSA, which makes it a lot more efficient.

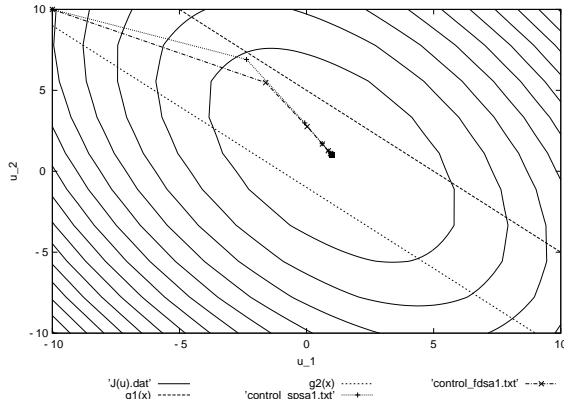


Figure 4: SPSA vs FDSA

Simple experiments with $p = 2$ showed that SPSA converges in the same number of iterations as FDSA. The latter follows approximately the steepest descent direction, behaving like the gradient method (see Figure 4). On the other hand, SPSA, with the random search direction, does not follow exactly the gradient path. In average though, it tracks it nearly because the gradient approximation is an almost unbiased estimator of the gradient, like shown in the following lemma found in [3].

Lemma 1 Denote by

$$b_n = E[\hat{g}_n | \theta_n] - \nabla J(\theta_n)$$

the bias in the estimator \hat{g}_n . Assume that $\{(\Delta_n)_i\}$ are all mutually independent with zero-mean, bounded second moments, and $E|(\Delta_n)_i|^{-1}$ uniformly bounded on \mathcal{U} . Then $b_n \rightarrow 0$ w.p. 1.

The detailed proof is in [3]. The main idea is to use conditioning on Δ_n to express $E[(\hat{g}_n)_i]$ and then to use a second order Taylor expansion of $J(u_n + c_n \Delta_n)$ and $J(u_n - c_n \Delta_n)$. After algebraic manipulations implying the zero mean and the independence of $\{(\Delta_n)_i\}$, we get

$$E[(\hat{g}_n)_i] = (g_n)_i + \mathcal{O}(c_n)$$

The result follows from the hypothesis that $c_n \rightarrow 0$.

Next we resume some of the hypotheses under which u_t converges in probability to the set of global minima of $J(u)$. For details, see [5], [3] and [6]. The efficiency of the method depends on the shape of $J(u)$, the values of the parameters a_k and c_k and the distribution of the perturbation terms Δ_{ki} . First, the algorithm parameters must satisfy the following conditions:

- $a_t > 0$, $a_t \rightarrow 0$ when $t \rightarrow \infty$ and $\sum_{t=1}^{\infty} a_t = \infty$; a good choice would be $a_k = \frac{a}{k}$, $a > 0$;
- $c_t = c/t^\gamma$, where $c > 0$, $\gamma \in [\frac{1}{6}, \frac{1}{2})$;
- $\sum_{t=1}^{\infty} (a_t/c_t)^2 < \infty$.
- Δ_{ti} must be mutually independent zero-mean random variables, symmetrically distributed about zero, with $|\Delta_{ki}| \leq \alpha_1 < \infty$ and $E|\Delta_{ki}^{-2}| \leq \alpha_2 < \infty$ a.s., $\forall i, k$. A good choice for Δ_{ki} is Bernoulli ± 1 with probability 0.5. The uniform and normal distributions do not satisfy the finite moment conditions, so can not be used.

The loss function $J(u)$ must be thrice continuously differentiable and the individual elements of the third derivative must be bounded: $|J^{(3)}(u)| \leq \alpha_3 < \infty$. Also, $|J(u)| \rightarrow \infty$ as $u \rightarrow \infty$.

In addition, ∇J must be Lipschitz continuous, bounded and the ODE $\dot{u} = g(u)$ must have a unique solution for each initial condition.

Under these conditions and a few others (see [5]), u_k converges in probability to the set of global minima of $J(u)$.

RESULTS

The experiments done with the ECOSIM simulator showed that effectively, SPSA outperforms the SQP method using the finite differences gradient approximation. We realized that the quality of the control was very much depending on SPSA parameters (in particular of the initial values of a , c and the value of γ), so we performed a few pilot simulations in order to find the right values. The fastest simulation did twice better than FDSA, i.e. the same experiment was

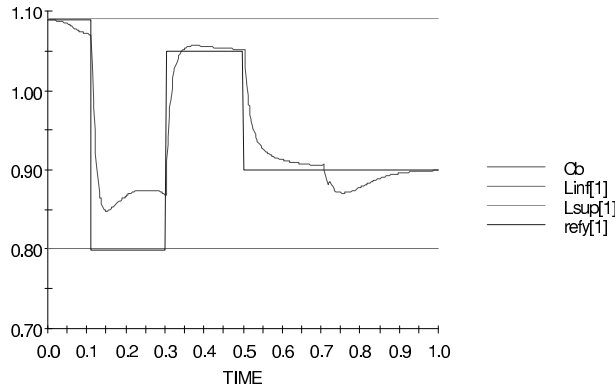


Figure 5: Concentration of Product B

two times faster when using SPSA. Like expected, the latter needed $p = 2$ times less cost function evaluations (in fact, the dimension of the problem here is 2: $u_t = (F, Q_k)$). The resulting control can be seen in Figure 5, where are plotted the reference level changing over time, the upper and lower bounds of the controlled variable c_B and the concentration of product B (y_t) itself. The small perturbation at time 0.7 is due to the change of temperature provoked on purpose. We see that the control is handling it well. Though, the controlled variable seems having some difficulties when near its upper and lower bounds, indicating that our way of treating the limit constraints may be incorrect. We used a dynamic stopping criteria of the form $\|\Delta u\| \leq \epsilon$, where $\epsilon = 0.1$ in most of the experiments, implying a small number of SPSA iterations. A higher precision was very costly, indicating that what we gained in speed was unfortunately lost in precision. However, we believe that a better choice of parameters can make the method more robust and this makes part of our on-going work.

CONCLUSION

In this research we present a methodology for global optimization adding noise to the observations of the gradients in order to achieve better performance of the model. Theory showed that the addition of random noise can make the control variables attain near optimality much faster than deterministic methods as confirmed by some of our experiments. In addition, this method can provably overcome the curse of dimensionality and thus be used in larger problems. However, finding the suited parameters proved to be challenging, showing that a particular attention should be given to this point.

References

- [1] D. P. Bertsekas. *Nonlinear Programming*. Mass. USA, athena scientific edition, 1999.
- [2] H. Chen and F. Allgower. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. In *Automatica*, 1998.
- [3] M. C. Fu and S. D. Hill. Optimization of discrete event systems via simultaneous perturbation stochastic approximation. *IEEE Transactions*, 29:233–243, 1997.
- [4] A. K. H. Chen and F. Allgower. Nonlinear predictive control of a benchmark cstr. In *Proceedings of 3rd European Control Conference*, pages 3247–3252, 1995.
- [5] J. L. Maryak and D. C. Chin. Global random optimization by simultaneous perturbation stochastic approximation. In *Proceedings of the American Control Conference*, pages 756–762, 2001.
- [6] J. C. Spall. An overview of the simultaneous perturbation method for efficient optimization. *John Hopkins APL Technical Digest*, 19(4):482–492, 1998.
- [7] C. D. P. W. Colmenares, S. Cristea and al. Mld systems: Modeling and control experience with a pilot process. In *Proceedings of IMECE 2001*, New York, USA, 2001.

ACKNOWLEDGEMENTS

The work of the first two authors was sponsored in part by NSERC and FCAR Grants of the Government of Canada and Quebec.

XML-BASED MODELLING LANGUAGE FOR TECHNICAL NETWORKS

Alexei Lisounkin
Gerhard Schreck

Fraunhofer-Institute for Production Systems
and Design Technology

Pascalstrasse 8-9
10587 Berlin
Germany

E-mail: {alexei.lisounkin, gerhard.schreck}@ipk.fhg.de

Pedro Pablo Alacórn Caveno
Juan Garbajosa Sopeña
Agustín Yagüe Panadero

Universidad Politécnica de Madrid
Escuela Universitaria de Informática

Ctra. de Valencia, Km 7
28031 Madrid
Spain

E-mail: {pcaveno, jgs, ayague}@eui.upm.es

KEYWORDS

Software engineering, object-oriented modelling, topology modelling, network language, model validation and verification.

ABSTRACT

The main issue of this paper is the XML-based modelling language for describing network-based technical systems. The modelling language was developed in order to make use of the advantages of the XML technologies for the parsing of model descriptions and the model processing. For this, applying the XML technologies will allow for an object-oriented interaction of a model based application via internet. Moreover, methods for model verification based on XML Schema mechanisms as well as by means of a formal interpretation of the model network structure were also designed.

The modelling language is developed within the framework of an European research and development project XNetMod "XML Based Modelling Language for Simulation of Technical Networks" (CRAFT-IST Contract No. 2001-52057). The use of new internet-based communication tools and standards will prompt the development of new services and thereby increase the efficiency of the business practices.

INTRODUCTION

The growing demands for a highly functional validity and safety of application software components for technical systems is increasingly being associated with the use of standard software engineering and development methods and tools. The adaptation and application of available standard software engineering methods and tools provide the required development efficiency and system validity. Here, such application software is considered, which substantially involves simulation-based supervision, control, and decision making mechanisms. This means that a simulation model of the technical system forms the functional core of the application software.

In the scope of the described application field, the standard

software engineering methods and tools can address the following aspects:

- development of an application specific functional simulation core, and
- implementation of the interchange mechanisms – interfaces – between the simulator and the environment.

Simulator Components

Highly sophisticated software tools for configuring numeric simulation tasks and the libraries for their numeric treatment currently exist and are being continuously improved. The latest developments are devoted to creating toolboxes that are specific to the application field. The configuration mechanisms and interfaces of these tools are still very different. A standard approach involving the XML technology appears to be a fruitful one.

Moreover, the description of technical component data in CAE-tools and *e*-business applications is increasingly provided involving the XML-based specification languages. The possibility to reuse the same data-bases for the configuration of the simulation software is also tempting.

Interface Components

Especially for components like communication modules, interface methods, and internal data model builder, the variety of implemented components offered is huge. Moreover, even on levels close to technological processes, the communication mechanisms incorporates more and more standard tools like Ethernet with TCP/IP protocol.

The configuration and monitoring interfaces of system terminal units and of core systems are being increasingly provided via internet-based tools and techniques – *e.g.* via interfaces based on web-browsers with Java Server Pages, data and functional interfaces via SOAP, XML languages with corresponding parsers. The configuration of the application dedicated software components will also be increasingly provided by using configuration languages and software generating tools. In all of these areas, the XML language family is playing an important role.

XML-BASED MODELLING LANGUAGES

The history of high level modelling languages is very exciting. Until now, the aspects of modelling languages had to be considered in close relation with tools which could provide a reasonable interpretation of the language. The reason is, that in their majority, modelling languages play the role of the configuration tool for the corresponding simulation interpreters. The semantic value of the languages was rather limited.

Only recently, under an impact of functional programming and object-oriented software technologies, the modelling languages have begun to involve not only the informational but also the semantic component (compare with *Modelica* modelling language, Fritzson 2002). A modelling language that specifically represents a technical system can serve as an information and knowledge carrier. The modelling data becomes the medium for the knowledge exchange (Jüngel 2000).

Considering this new role of modelling languages, the language interpreter should become more flexible with respect to the grammar of the language. The new challenge of the language interpreters is to extract a relevant (from the interpreter's point of view) knowledge from a modelling language representation and to process it.

From this point of view, an XML-based language has dramatic advantages as a language for data interchange and application interaction. Moreover, we can emphasise that technical systems with network architecture are predetermined for distributed supervision and automation solutions. Here, the application of internet technologies is extremely efficient.

The reasons for an increased usage of XML technology are obvious: low costs for the development and high stability of the software solutions due to open source license concepts and a high number of technology users.

The main focus of our work is the development of the XML language family for modelling of the application kernel in model-based simulation and control systems. By that we connect a clear data architecture of the application model with some formal rules for the model structure and model parameter verification, and we profit from available XML interpreting software.

A SYSTEM WITH NETWORK ARCHITECTURE

A network structured system possess the following important properties:

- The processing elements of the system construct the nodes of the network architecture and belong, with respect to their types, to a rather limited set of so called "standard processing elements".
- The topological relations between the processing elements are the links of the network and represent some kind of functional relations between the nodes.

- The network structure may possess a special (pattern based) structure (trivial example – Petri net).
- The connected processing elements must match some specification with respect to the structure and values of their attributes.

The set of processing elements included in the system constructs the core of the system data model. The topological (or functional) relations between the elements are considered as references between the nodes. It is important to emphasise that advanced simulation-based tasks may evaluate the node relations of different nature – technology-caused connections, supervision/control reasoned data flow, resource logic relations *etc.* An allied processing of relations, which may have different types and nature, is increasingly important for the advanced modelling technologies.

With respect to the simulation task context and functional content of the networks, the relations can be interpreted in different ways depending on the interpretation scheme of the network structure. In an analogous manner, respectively to different interpretation schemes, the model of the system can be processed for the verification of the model structure or parameterization.

It is reasonable to consider the model structure information – model topology representing different types of relations between functional elements – in an integrated way. The development of unified mechanisms for the model structure and data verification is also closely related to this task.

Model Verification Mechanism

Obviously, the system topology correctness is crucial for the processing of simulation-based tasks. Therefore, special efforts were made in order to elaborate proper methods for the network structure validation. The developed approach includes two mechanisms:

- use of XML specific tool – XML Schemas – for verification of model elements and topology structure,
- use of rules which define allowed and not allowed model patterns for verification of the model structure.

For the XML-Schema-based mechanism, the definition of sophisticated XML Schemas for model components and relation between them must be provided. The verification procedure can be provided by available XML parsers, for example by Xerces XML parser developed by the Apache Software Foundation (<http://www.apache.org/>).

The rule-driven verification of the model structure allows to include a set of model verification rules in the model description and apply them to the model parts by means of a specialised interpreter. Here we consider two main aspects of the verification:

- rule-based verification of model node parameter templates and values, and
- rule-based model structure verification involving an algebra approach.

DATA MODEL AND XML REPRESENTATION

The mathematical structure of the object can initialize the following data model (see Figure 1). Here, the class "model" is the root class for the entire model. The classes "nodeLevel1" and "nodeLevel2" represent node elements of different types or even hierarchical levels. The number of hierarchical levels is not limited. For the hierarchical organisation of the model structure, a referencing between node elements of different levels is allowed. This referencing should build a clear tree structure, which will induce a clear structure of the interpreting mechanisms.

As the model can possess several net topologies defined over the node elements, it is allowed to define net topologies of different types in the same model. The networks are represented by classes "net". The "pin" elements refer to the nodes which are directly connected by means of the related network. Additionally, "rule"-elements, which may define legal and illegal network patterns for the relevant network, are also included in the model.

The design of advanced modelling and configuration languages is being considered with respect to the approach of generic modelling and reusable software. For us, development of an XML-based language for describing the network topology in conjunction with application line specific XML languages for describing the node characteristics is considered an appropriate technology. Herewith, the network modelling and verification language has an integrating role. The fact that the scanning, parsing, and interpreting tools for XML-based languages are available justifies the use of XML grammar.

Language Structure

In general, XML-based grammars are appointed to represent data models with a tree structure. Obviously, the depicted class model can be also depicted by means of an XML-based language.

Using the natural structure of the XML grammar, the model structure as it is shown in Figure 1 can be easily transformed in a form with four main semantic parts:

- a set of functional elements of a network ("node"-elements) – nodes in a graphical representation of the model,
- a set of networks ("net"-elements) with their connections ("pin"-elements) – edges in graphical representation of the model,
- a set of verification rules ("rule"-elements) which may define legal and illegal network patterns for the relevant network, and
- a set of attributes ("attr"-elements) related to node elements, network definitions, and connection descriptions.

The modelling language provides a rather conservative set of conventions for the definition of the set of "node"-elements and the topology of the related networks. Addi-

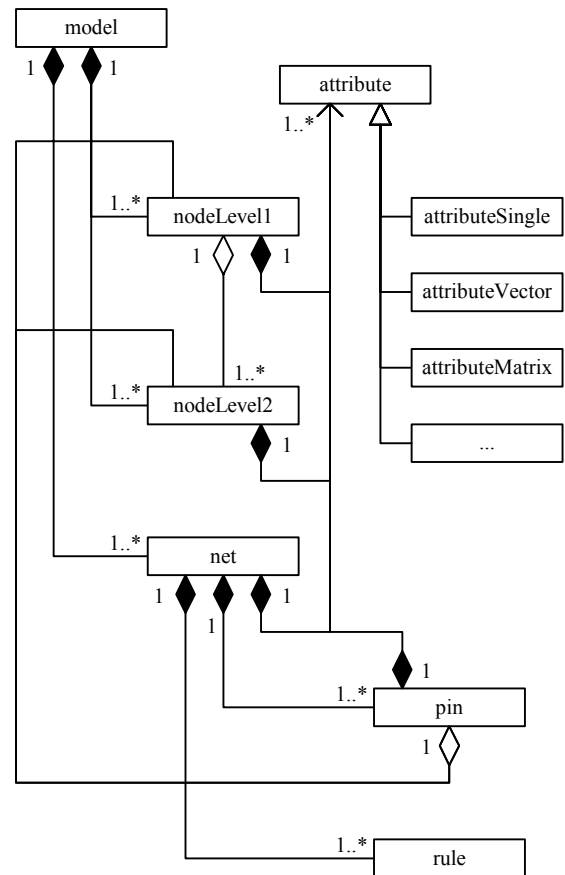


Figure 1: Generic class model of a network structure.

tionally, rules for the verification of the network structure involving pattern-based mechanisms ("rule"-elements) will be also defined. Even at this stage, without definition of the "node"-elements internal structure, the model topology can be interpreted in a formal way involving the patterns defined by the "rule"-elements.

The definition of the internal structure of the "node"-elements and the "attr"-elements can be provided in an external file. This possibility can be exploited in cases when the origin of "node"-elements is foreign – the information about the elements is provided by an element supplier, the model engineering task may access to external data bases. Herewith, local variations of the nodes' functionality specification result only in a local adaptation of the network model. Moreover, the separation of the model elements description and network definition allows a rather independent access to the model functional elements and the model topology on the model processing level.

Summarising we want to emphasise that the network modelling language defines the structure – topology – of the technical facility or process and provides the highest abstraction level for the description of the process functional relations. Therefore, the modelling language possesses the highest level in the language hierarchy.

Let us consider the example of a technical network depicted in Figure 2.

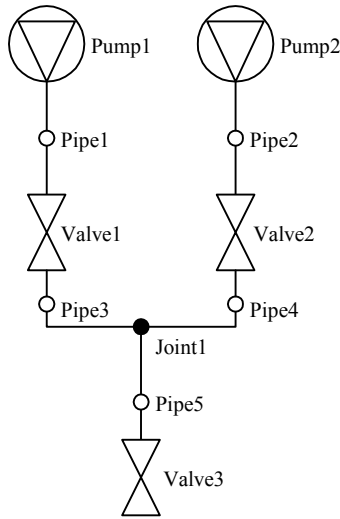


Figure 2: Example of a water distribution network.

The corresponding XML representation of the model can be as follows:

```
<model name="water network">
  <!-- system nodes with attributes -->
  <node name="Pump1" type="pump">
    <pressure unit="hPa">1.1</pressure>..
  </node>
  <node name="Pipe1" type="pipe">..</node>
  <node name="Valve1" type="valve">..</node>
  <node name="Pipe3" type="pipe">..</node>
  <node name="Joint1" type="joint">..</node>
  <node name="Pipe5" type="pipe">..</node>
  ..
  <!-- system network with connections -->
  <net name="flow">
    <pin from="Pump1" to="Pipe1"/>
    <pin from="Pipe1" to="Valve1"/>
    <pin from="Pipe3" to="Joint1"/>
    <pin from="Pipe4" to="Joint1"/>
    <rule name="pump*valve" type="pattern"
      from="pump" to="valve"/>
    ..
  </net>
  ..
</model>
```

XML-SCHEMA BASED MODEL VERIFICATION

Using XML technology, the model verification can be provided from a syntactic and, partially, from a semantic point of view (Van der Vlist 2002). The well defined language structure offers the possibility to localize the syntactic errors. The semantic aspects of the model verification can be included in the XML Schema templates related to the XML document. By means of XML Schema, the verification of element structure and attribute type correctness are feasible. Moreover, the verification of the element name uniqueness as well as the legality of references can be also provided by the XML-Schema specification features.

Thus, by means of XML Schema definition and use of "key"-element it is possible to declare some attributes or elements of the XML document as primary keys. Once a primary key has been defined, references to the key can be established by means of a "keyref"-element.

This feature is used for the referencing of the nodes in the framework of the pin elements. The next XML Schema code shows how to define the "name"-attribute of "node"-element as primary key for use to address elements of the network:

```
<xs:key name="keyNode">
  <xs:selector xpath="./node"/>
  <xs:field xpath="@name"/>
</xs:key>
```

Referencing of "node"-elements within "pin"-elements by use of the "keyref"-element will be defined, like in the XML Schema code for "net"-element below:

```
<xs:element name="net">
  <xs:complexType><xs:sequence>
    <xs:element name="pin"><xs:complexType>
      <xs:attribute name="from"
        type="string" use="required"/>
      <xs:attribute name="to"
        type="string" use="required"/>
    </xs:complexType></xs:element>
    <xs:keyref name="krf" refer="keyNode">
      <xs:selector xpath="pin"/>
      <xs:field xpath="@from"/>
    </xs:keyref>
    <xs:keyref name="krt" refer="keyNode">
      <xs:selector xpath="pin"/>
      <xs:field xpath="@to"/>
    </xs:keyref>
  </xs:sequence></xs:complexType>
</xs:element>
```

FORMAL METHODS FOR MODEL VERIFICATION

Based only on the SML Schema functionality, it is not possible to verify the semantic of XML documents. For example, it is not possible to verify the matching of corresponding parameters in connected "node"-elements. Moreover, for an advanced verification of a network model, verification of the network structure is extremely important. Referencing the example of the water distribution network, we can require that every pump must be sealed by a valve, although between the pump and the valve several pipe elements are allowed. Here, the verification of the network structure means the analysis of patterns in the network structure.

Therefore, the aim of the formal verification methods is to provide a model verification mechanism based on clearly defined verification rules. The verification will be performed by specially developed parsers. Here we consider two approaches:

- the verification of network configuration and
- the verification of network structure.

The essential idea of the first approach is to find the correspondence or match for key attributes for nodes that are connected respectively. The matching can be provided on the following levels:

- Attribute instance check provides verification of attributes, relevant for the direct connection of two nodes. That means, that for a given pair of nodes, a

specified attribute must belong to both nodes. We will denote the relevant attributes as the connection key attributes.

- Attribute value check provides verification of attribute values, relevant for the direct connection of two nodes. That means, that for a given pair of nodes, a specified attribute must belong to both nodes, and the actual values must be equal.
- Attribute constant check provides verification of attribute values, relevant for the direct connection of a set of nodes. That means, that for a given set of nodes, a specified attribute belongs to a node, its actual value must be equal to a pre-defined constant.

As already mentioned, the main idea of this technique is the application of algebraic and pattern based methods for the model verification. Respectively, an algebra definition should be provided for the verification of network properties. Moreover, the definition of several algebraic systems could be needed, if different verification patterns are to be applied. Here, the use of network algebra theory could be helpful (Stefanescu 2000).

The implementation of this approach includes three steps:

- the development of the language for the pattern specification,
- the development of the translation mechanism for the mapping of a network topology into a proper algebraic term, and
- the development of the interpreter for the algebraic term evaluation

The language elements for definition of pattern structures was also developed. Thus, a rule for the verification of the pump-pipe-...-pipe-valve sequence can be assigned to the related net and defined as follows:

```
<rule name="pump*valve" type="pattern"
  from="pump" to="valve"/>
```

The theoretical aspects of the mapping of a network topology into a proper algebraic term are also elaborated in (Stefanescu 2000).

MODEL INTERFACE

A well organised data model provides a good background for an efficient implementation of the model data access. The functional aspect of the data access can be put into effect using methods of functional programming (Thompson 1991). The implementation of the interface can be made generic, independent from the subject field.

In accordance with this approach, the communication can be provided on the message basis, the format of the messages can be chosen in accordance with the XML-based modelling language. In this case, the same XML-parser/interpreter can be used for the model construction within the memory and for the data exchange with the model. For the tests, Xerces libraries developed by the Apache Software Foundation (Xerces, 2003).

CONCLUSION AND FURTHER RESEARCH

The developed XML-based language was applied for the solution of modelling tasks with respect to facility behaviour. The first application of this technology is devoted to the modelling of water distribution networks. The further applications will be the modelling of other physical media distribution networks (oil and gas networks), the modelling of document flow processes with data capturing procedures, and physical resources (human teams and vehicles) distribution modelling related to a road network. Such different application areas was chosen in order to prove the applicability of the developed approach for a rather wide application spectrum.

Further research will be also focused on the development of further algebraic structures for verification of network topology. Here, the mentioned application areas will provide the semantic basis for the design of the algebras.

ACKNOWLEDGEMENTS

The language with the represented model verification mechanisms is being developing in the framework of an European research and development project XNetMod "XML Based Modelling Language for Simulation of Technical Networks" (CRAFT-IST Contract No. 2001-52057). The consortium includes two RTD performers – *the Fraunhofer IPK* (Berlin / Germany) and *the Universidad Politécnica de Madrid*, Escuela Universitaria de Informática (Madrid / Spain). The first users of the technology are four industrial companies – *ELPRO Prozessindustrie- und Energieanlagen GmbH* (Berlin / Germany), *SimPower Simulator Systeme GmbH* (Karlstein am Main / Germany), *Investigacion y Programas S.A.* (Madrid / Spain), and *ACEBO, C.B.* (Arcas de Villar - Cuenca / Spain).

REFERENCES

- Fritzson P., Aronsson P., Bunus P., Engelson V., Saldamli L., Johansson H., and Karström A. 2002. "The Open Source Modelica Project." In *Proceedings of the 2nd International Modelica Conference* (Oberpfaffenhofen, Germany. March 18-19). <http://www.Modelica.org/Conference2002/papers.shtml>.
- Jünger M., Kindler E., Weber M. 2000. Towards a Generic Interchange Format for Petri Nets. In: Bastide, R. et al (eds.): *Meeting on XML/SGML based Interchange Formats for Petri Nets*, 21st ICATPN. (Aarhus, Denmark, June). <http://www.informatik.hu-berlin.de/top/pnml/download/XML-Position00.ps>.
- Stefanescu G. 2000. *Network Algebra*. Springer-Verlag. Berlin Heidelberg.
- Thompson S. 1991. *Type Theory and Functional Programming*. Addison-Wesley.
- Van der Vlist E. 2002. *XML Schema. The W3C's Object-Oriented Descriptions for XML*. O'Reilly.
- Xerces - XML parsers in C++. version 2.0.0. 2003. <http://www.apache.org/>

AUTHOR LISTING

AUTHOR LISTING

Abdullah R A.	188	Dalton J.	83
Acebes L.F.	43	Dangelmaier W.	233
Adabala N.	502	de Arruda L.V.R.	346
Ai X.	473	de Prada C.	43/533
Alacórn Caveró P.P.	539	Dedoncker P.	149
Albanese M.	118	di Gravio G.	253
Alegre J.A.	28	Dvornik J.	192/336
Ali B.	341		
Alimi A.	317	Ecker H.	180
Alves R.	43	Ezquerria J.M.	238
Andersson J.	59		
Aracil M.-J.	442	Fei G.	502
Argibay-Losada P.J.	68	Fernández M.	167
Arjona E.	125	Ferrarini L.	88
Armesto L.	362/367	Fogelholm J.	295
		Fontana M.	496
Baccouche M.	418	Franek F.	48
Bagheri A.	387	Franke H.	233
Baines T.	326	Fuster S.	367
Baltcheva I.	533		
Barone S.	217	García I.	141/401
Bartkevicius S.	131	Ghandia G.S.	175
Bassanini M.	253	Gicquel R.	53
Benabdelhafid A.	317/418	Görmer J.	99
Benaissa M.	317	Gorokhov A.	136
Bennahammed K.	149	Gracia L.	357/362
Bertotto C.A.	312	Grenez F.	149
Boukachour J.	418	Griffin B.	321
Bréholée B.	413	Grillinger P.	206
Bruha I.	48	Guzmán D.	167
Bruniaux P.	523		
Bueno G.	125	Hadfield L.	326
Bus N.	53	Hands P.S.	511
Byun H.-S.	211	Heo S.-N.	211
		Hollauer C.	154
Carpanzano E.	290	Honorati L.	253
Casas S.	167	Hoskote S.	477
Castelain J.-M.	523	Hössinger A.	159
Castillo J.-I.	442	Houhamdi Z.	62
Cataldo A.	290	Huang X.	491
Ceric H.	154	Huyet A.-L.	113
Chen T.	491		
Chen X.	473/483/519	Jiménez F.	367
Choi Y.-H.	211		
Conti C.	201	Kacha A.	149
Costantino F.	253	Karst D.	457
Crappe J.	201	Kay J.M.	326
Cristea S.	533	Khatib C.	99
Cugini U.	496	Khoshnejad K.	387
Curcio A.	217	Kim S.-T.	211

AUTHOR LISTING

Kim Y.K.	477	Oleynik A.	136
Kindler E.	263	O'Maloney T.	321
Koch W.	429	Orsulic M.	192
Koncar V.	462	Osmólski W.	437
Kouroussis G.	201		
Kowalska Z.	397	Panadero A.Y.	539
Křivý I.	263	Paris J.-L.	113
Krus P.	59	Park K.H.	462
Kunicina N.	243	Pastijn H.	24
		Pearce T.	38
Ladbrook J.	326	Pereira C.E.	312
Landaluze J.	167	Perles A.	103
Lawrence P.	331	Pla-Castells M.	141/401
Lee Y.T.	307	Pranevicius H.	271
Levchenkova A.	243/247	Putilov V.	136
Li L.	38		
Lian Y.-H.	258	Rafik T.	73
Lisounkin A.	539	Rajabally E.	83
Lopez-Ardao J.C.	68	Ramanan B.S.	511
Lopez-Garcia C.	68	Ramos A.E.M.	346
López-Valpuesta L.	442	Rapp W.A.	457
Luo Y.	307	Reuter M.	380
		Rizzi C.	496
Macerauskas V.	131	Rodriguez-Rubio R.F.	68
Maciejewski M.	437	Romanin-Jacur G.	467
Maffezzoni C.	88		
Magnenat-Thalmann N.	502	Santonja V.	28
Makihara K.	180	Sarkauskas K.	131
Marti A.	103	Scheideler P.	233
Martínez A.	167/238	Schiavo F.	88
Martínez R.J.	141/401	Schreck G.	539
Maruri L.	238	Selberherr S.	154/159
Marx Gómez J.	99	Sen P.	83
Mason S.	326	Serrano J.J.	103
Matta A.	118	Shamey R.	451
McLean C.	307	Shin J.-H.	211
Mebrahtu H.	299	Simundic S.	336
Merino A.	43	Siron P.	413
Mileham T.	299	Smyshlaev I.	247
Miller J.	21	Sopeña J.G.	539
Molero X.	28/103	Stoian V.	392
Mora M.C.	223	Suarez-Gonzalez A.	68
Munitic A.	192/336		
		Tan X.	483
Narbuntowicz E.	276	Tanguy A.	263
Nariman-Zadeh N.	387	Teijeiro-Ruiz D.	68
Nassehi V.	341	Tolio T.	118
Neuhöfer J.A.	375	Tornero J.	223/362/357/367
Neves Jr., F.	346	Torregrosa I.	28
Normey-Rico J.E.	43		
Núñez L.	167/238		

AUTHOR LISTING

Van Landeghem H.	258	Yang Y.	457
Van Loock R.	24		
Van Utterbeeck F.	24	Zhang H.	519
Van Welden D.F.	5	Zhao X.	451
Vazquez–Abad F.J.	423/533	Zottolo M.	285
Vercouter D.	201	Zubieta L.	423
Verlinden O.	201		
Vidarte A.	238		
Villaverde D.	167		
Vosoughi M.	451		
Wagner F.R.	312		
Wainer G.A.	38		
Walerjańczyk W.	276		
Walker R.	299		
Wang H.	519		
Whittle S.	83		
Williams E.J.	285		
Winchester S.C.	511		
Wittmann R.	159		