SCIENTIFIC PROGRAMME
SIMULATION METHODOLOGY
SIMULATION METHODS AND TECHNIQUES
USING TIMED PETRI NETS IN DISCRETE SIMULATION

Jaroslav Sklenar
Etienne Caruana
Department of Statistics and Operations Research
University of Malta
Msida MSD 06, Malta
Web: http://staff.um.edu.mt/jskl1/
E-mail: jaroslav.sklenar@um.edu.mt, etienne.caruana@gov.mt

KEYWORDS
Petri Nets, Timed Petri Nets, Discrete Simulation,
Manufacturing Systems.

ABSTRACT

Petri Nets offer a graphical language for both analytical and simulation modeling of complex and concurrent systems. The paper deals with simulation methodology based on application of timed Petri nets. After a brief introduction to timed Petri nets, their special form used by the PetriSim simulation package is introduced. Simulation technique based on the description of the simulated system by a timed net is first studied generally, then the results are shown by the simulation of a beverage-line manufacturing system. Though the timed nets used are specific for PetriSim, the results can be applied, possibly with minor modifications, for all types of timed Petri nets based on the firing duration paradigm.

INTRODUCTION

There are various methods that can be applied in the analysis of concurrent and distributed systems, like for example queuing networks, various process algebras, and various modifications of Petri Nets (PN). Use of PNs offers the following (Desrochers 1995):

- It captures precedence relations and structural interactions of stochastic, concurrent and asynchronous events
- Its graphical nature helps to visualize complex systems
- Conflicts and buffer sizes may be modeled easily
- Deadlocks may be detected easily
- It has well developed practical and mathematical foundation
- The various extensions of PNs allow for both qualitative and quantitative analysis
- There are supporting software packages.

PNs were developed into high-level PNs in order to extend their applicability in the performance analysis of various systems. The advance in this subject was mainly attributed to the introduction of color, hierarchy and time to the basic PN model. The main reason of extending the basic PN definition with color and hierarchy was to reduce the structural complexity of PN models. The introduction of time to the PN semantics, on the other hand, was necessary for the proper performance analysis of studied systems by associating time with all the possible events occurring during execution. The basic PN structure of timed PNs is however unaltered. The analyst still has the possibility to derive the underlying PN structure of the model on which a series of structural analysis could be performed, in order to analyze the structural and behavioral properties of the system. In this paper we assume that the reader is familiar with basic ideas of PNs, at least with their basic version called today Place/Transition (Pl/Tr) nets. For basic ideas see for example the introductory article (Reisig and Rozenberg 1998), for more details (Rozenberg and Engelfriet 1998) or (Desel and Reisig 1998).

There are various ways of introducing time into PNs. In general, time may be associated with places, transitions, tokens, and arcs. We believe, that from the modeling point of view the most natural way is the one that associates time with firing of transitions. In real systems it is the activities that take time and consume other resources. In PNs activities are modeled by firing of transitions. When associating time with transitions, it is also necessary to define the firing semantics (atomic firing or firing in three stages), the conflict resolution policy (in case of more enabled transition) and the nature of the firing delay (deterministic, random). That’s why there can be many different types of timed PNs with firing delay concept. For an introduction see the paper (Ajmon Marsan et al. 1998) where the authors define the so called Generally Distributed Timed Transitions Stochastic Petri Nets (GDTT_SPN) as a general class of timed PNs in which time is associated with transitions. GDTT_SPNs are then specialized into various types like for example Timed Deterministic Petri Nets (TDPN), in which a constant firing time delay is attached to all transitions in the net, Stochastic Petri Nets (SPN), in which all the transitions have exponentially distributed firing delays, Generalized Stochastic Petri Nets (GSPN) which are SPNs extended with immediate firing transitions, etc.

Timed PNs can be applied in two different ways. First there are timed nets with analytical capabilities. For example SPNs and GSPNs (and their other modifications) can be analyzed as Continuous Time Markov Chains. This topic is dealt with for example in (Donatelli and Franceschinis 1998), (Haverkort 1998) and (Lindemann 1998). Use of analytical capabilities of timed PNs is limited by the well known problem of very fast growth of
the state space and from the modeling point of view by strong, in fact unrealistic, assumptions of most models, like for example exponentially distributed firing delays. This paper deals with the use of timed nets in simulation. Time net is used as a descriptive language during creation of the simulation model. Performance parameters are then obtained by performing simulation experiments.

**PETRISIM NETS**

PetriSim is a tool for discrete simulation based on the use of timed PNs. It contains a user-friendly graphical editor to create a net that can then be simulated in various modes, either as a PI/Tr net without time or as a timed net. The PI/Tr mode is used to check the behavior of the net and to search for deadlocks. Timed mode is used to obtain various performance measures. PetriSim is a simulation tool, so it does not support analysis of the underlying Petri net. For more details about PetriSim and its use visit its homepage at [http://staff.um.edu.mt/jskl/petri_sim/](http://staff.um.edu.mt/jskl/petri_sim/) that also contains an on-line manual.

**Timing**

PetriSim nets are based on the firing delay approach with three stages firing semantics. Conflicts among enabled transitions are so far resolved by random selection, all enabled conflicting transitions having the same probability. Firing delays are generated by user cope snippets in Turbo Pascal 7, so they can be constant, deterministic (dependent for example on the network status) or random with arbitrary distribution. The mechanism is as follows. When the firing starts, the tokens are removed from input places and then the so-called starting code snippet is performed. Typically it generates the firing delay. When firing ends, the so-called ending code snippet is performed and then the tokens are added to output places. Mostly ending snippet is not used. By default, all transitions are immediate ones with zero delay and without code snippets. Code snippets are written by the user when the net is created by the graphical editor and typically they contain just few statements, mostly only one procedure call to generate the delay.

**Generalized Arcs**

Inhibitor arcs have been defined in order to overcome the inability of PI/Tr nets to test for zero marking. PetriSim has generalized inhibitor arcs that enable the transition if the marking is less than the multiplicity of the arc – inversion of the normal arc’s test. Third type of arcs, the so-called generalized test arcs, are used to simplify nets’ graphs. Testing arcs just test the presence of tokens; during firing the tokens are not removed. For immediate transitions a test arc replaces a pair of arcs with opposite directions. All types of arcs are shown in Figure 1.

**Types of places and transitions**

PetriSim places and transitions have types that, with one exception, just affect the screen icons. The default type is the standard circle for places and the thick bar for immediate transitions – see Figure 1. Other types are shown in Figure 2. So far the only other types of places are the horizontal and the vertical queues. For transitions there is the rectangle for timed transitions, icon for generators of tokens (shown in Figure 2 together with the enabling place) and branching. Branching transitions have different semantics caused originally by the need to model movements of customers in queueing networks. For a branching transition a firing starts as usually. When the firing ends, the tokens are placed to one output place only. By default the output place is chosen randomly, all with the same probability. This can be changed by the ending code snippet. Transitions have the property branch set to the default zero. Storing other value, that is interpreted as the output arc number, can model deterministic or other than uniform random branching.

**PETRISIM NETS AND DISCRETE SIMULATION**

There are three ways of time control in discrete simulation. Activity oriented simulation; also called Interrogative Scheduling is not based on explicit scheduling of future events. For each activity in the model a condition is specified. If the condition is true, the activity is performed. For this way of viewing the dynamics of the system the use of PNs is very natural and straightforward. Transitions represent activities, marking of input places are the activity starting conditions, marking of output places are the results of the activity, that can be starting conditions of other
activities, etc. See Figure 3. Simulation methodology in this way involves definition of activities together with their conditions and results. Compatibility with PNs ideas is obvious.

![PN model of a conditioned activity](image)

Figure 3: PN model of a conditioned activity

The other two ways of time control are the event-oriented and the process-oriented simulation. When using time nets, the common problem is the simplicity of the timed nets graphical language compared with various semantically rich statements of discrete simulation languages. The simplest case is the scheduling of an event. It can be implemented by a timed transition; the event is then the end of the firing. The problem is cancelling of an event. Currently PetriSim does not support preemption, so a firing that has started cannot be stopped. A partial solution is shown in Figure 4. The only timed transition is d, the event itself is the firing of the transition e. Firing of the transition Cancel cancels the event by removing the token from the place confirm, the transition clear sets the initial status. Marking of the place c_c is the cancelling condition.

![Scheduling and cancelling of an event](image)

Figure 4: Scheduling and cancelling of an event

For the process-oriented way, the timing of the process has to be expressed as a sequence of events. There is no problem with synchronization and communication among processes because this is natural when using PNs. For example in Figure 3 the places c_1 to c_n can represent certain points reached by n corresponding processes. The transition Activity is then the synchronization point, because it cannot fire until all these places get at least one token.

### Example Simulation

The system under study is a soft drink manufacturing system. The main function of this system is the filling of the glass bottles. Figure 5 illustrates the main processes that are identified within this system.

![Activity diagram of the studied system](image)

Figure 5: Activity diagram of the studied system

The source of unclean glass bottles in plastic crates on pallets is referred to in Figure 5 as Store 1. Every pallet contains 40 crates and each crate contains 24 bottles. The Depalletiser un-stacks the crates from the pallets and introduces them to the Unpacker. This device separates all the unclean bottles from the crates. This operation starts two parallel processes by introducing 24 unclean bottles into the Bottle line and 1 unclean crate into the Crate line. The Bottle line consists of three main components, which are the Bottle washer, the Filler, and the Labeller with obvious functions. The Crate line is composed of the Crate washer. The Crate line and the Bottle line join in a synchronization process at the Packer. The Packer packs 24 clean and filled bottles from the Bottle line into one clean crate from the Crate line. All the crates produced are then stacked on an empty pallet by the Palletiser and transferred to Store 2. Figure 5 marks the percentage production rate for every component with respect to the filler that is the slowest component. Between the devices bottles and crates are moved by various conveyor belts marked in Figure 5 by symbols B_i and C_c. At the end of each conveyor, there are sensors, which stop the conveyors if they become full of bottles or crates respectively.
Building the model

The system in Figure 5 was converted into a timed net. Due to the nature of the system, deterministic time delayed transitions were used to model the sequential processing of bottles and crates within the main system’s components. The time delays were obtained by measurements on the real system. Places were used to represent queues with tokens representing bottles and crates. A separate exercise was conducted in order to estimate the speeds and the buffer sizes of the conveyor belts. As crates have the rectangular shape, belts capacities were simply computed by dividing the belt length by the crate’s size. Similarly for single line bottle conveyor belts the capacity was computed by dividing the length of the belt by the base diameter of the bottle. The maximum number of bottles that could reside in a multilane conveyor belt was computed by dividing its total area by the area taken by one bottle. The linear speeds of conveyors were calculated by measuring the time that a bottle(crate) took in order to move a known displacement assuming that the crates and bottles move at the same speed as the conveyor belts.

The timed net that models the system in Figure 5 is made of 42 places and 40 transitions. As it is made of just few repeating subsystems, only a typical, most complicated part representing a conveyor belt will be shown. All transitions are timed, even though they are drawn as thick bars. Figure 6 shows a timed net model of a conveyor belt, here it is the belt C10 before the Palletiser.

![Diagram of conveyor belt model](image)

Figure 6: Timed net model of a conveyor belt

A conveyor belt is represented by three places and two transitions. Places with the prefix i represent the crates (bottles) introduced at the beginning of the conveyors. Places with the prefix q represent the number of crates (bottles) queued at the end of the conveyor. Places with the prefix c(h) represent the total number of crates (bottles) inside the conveyor. Firing of transitions with the prefix t represents the arrival of a crate (bottle) at the beginning of the queue, while firing of transitions with the prefix u represents the departure of the last crate (bottle) at the far end of the conveyor. In Figure 6 firing of the transition uc9 represents the insertion of a crate into the conveyor belt C10 from the conveyor belt C9. With this firing, the places ic10 and cc10 receive a token. A token in ic10 means that a crate is being transported through the conveyor C10. The placeqc10 represents the queue that starts residing at the end of this conveyor. The number of tokens inside the place cc10 indicates the total number of crates inside this conveyor belt. An inhibitor arc from cc10 to the input transition uc9, with weight that is equal to the capacity of C10, has been introduced in order to stop the conveyor C9 as soon as C10 is full. Two timed transitions were needed for each conveyor in order to express the behavior of the crates. Firing of transition tc10 represents the arrival of a crate at the end of the queue, firing of the transition uc10 represents the movement of the first crate from the queue for processing. A deterministic delay has been assigned to all the transitions of the type as uc10. This delay corresponds to the time that is elapsed for a displacement, equivalent to the length of a crate, at the end of the queue. The delay time for tc10, however, had to be a function of both the speed of the conveyor belt and the length of the queue at the end of the conveyor as follows. Given that the net is in marking $m$, then $\tau_{uc10}$ denotes the firing delay of the transition tc10 at marking $m$:

$$\tau_{uc10}(tc10) = \frac{l_{c10} - l_{c}(mqc10)}{m(ic10)_{sc10}}$$

where $l_{c10}$ denotes the length of the conveyor belt C10, $l_{c}$ denotes the length of the crate and $sc_{c10}$ denotes the speed of C10. $l_{c10} - l_{c}(mqc10)$ gives the remaining distance that needs to be traveled by all the crates in ic10 in order to arrive at the end of the queue, at the marking $m$. The delay is hence obtained by dividing this distance by the linear speed of the conveyor C10, that is, $sc_{c10}$. This fraction has also been divided by $m(ic10)$ since, in practice, all the crates that are introduced on these conveyor belts travel in groups of size $m(ic10)$. The place $p_{t}$ is used to describe the behavior of the Palletiser, which processes crates sequentially. An inhibitor arc stops uc10 if a pallet is being processed. The principles applied to model the movement of the bottles within the Bottle line were similar with some alterations in the delay procedures and in the multiplicities of the arcs in the sub-nets modeling the multilane conveyors.

Simulation

The model was built in PetriSim and run for an 8-hour (one shift) simulation time. PetriSim automatically gathers the following parameters:

- Average, maximum and minimum marking
- Average, maximum and minimum waiting time of tokens inside each place
- The final marking.

Transitions:
- Mean, maximum and minimum firing time
- Total number of firings
- Utilization (portion of time when firing was on).

RESULTS

Table 1 contains utilization computed by PetriSim for all the transitions representing the main system’s components.
Table 1: Utilization of system’s components

<table>
<thead>
<tr>
<th>Component</th>
<th>Util.</th>
<th>Component</th>
<th>Util.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labeller</td>
<td>0.78</td>
<td>Packer</td>
<td>0.66</td>
</tr>
<tr>
<td>Filler</td>
<td>0.83</td>
<td>C. washer</td>
<td>0.64</td>
</tr>
<tr>
<td>B. washer</td>
<td>0.79</td>
<td>Unpacker</td>
<td>0.67</td>
</tr>
<tr>
<td>De-palletiser</td>
<td>0.59</td>
<td>Palletiser</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Since all components with high utilization reside within the bottle line, this is actually the bottleneck of the whole system. The analysis of the crate lines showed a rapid accumulation of crates on most of the conveyors within this line. Examples of these conveyors were C2 and C5, which demonstrated a high average number of crates of nearly 100% of their capacities. The conveyors that have not been constantly filled were C7 and C10, which had a 25% crate occupation. All this indicates room for improvement within the processing of the Crate line, which may be accomplished by increasing the speeds of the motors or by increasing the length of the fully occupied conveyors. A completely different behavior from the above has been demonstrated by most of the conveyor belts within the Bottle line. In fact, most of these conveyors have been nearly empty during the whole process. This could be attributed to their very fast motors. Also, the large widths of the multilane conveyors, which permitted the transportation of very large groups of bottles contributed to this result. The only accumulation of bottles resulted in conveyors B10 and B11, which seem to have been completely filled during the whole processing time. This accumulation of bottles may be explained by the performance of the Labeller, which doesn’t seem to have a processing rate fast enough to tackle all the arriving bottles. The production of this system, during its 8-hour shift, amounted to 6,583 crates, with an estimated average production rate of 19,749 bottles per hour. The discrepancy from the measured production rate value given from the company’s past records, which was of 19,000, is less than 4%. The above summary is only an overview of some of the results that could be obtained regarding the system’s performance.

**CONCLUSION**

Use of timed nets in simulation can bring some advantages when compared with other approaches to discrete event simulation. First there is always the possibility to analyze the underlying Petri (not timed) net in order to study the structural and behavioral properties. In this way it is possible for example to detect deadlocks caused by errors in the design. Another possibility is the detection of unbounded places that may cause overloading of buffers in the real system, etc. When compared with programming simulation models in discrete simulation languages, use of timed nets can simplify and speed up the whole process. Complex relationships among entities of the simulation model, that have to be programmed by specialized statements of the simulation language, are created by the execution of the timed net. When compared with visual simulation environments, like for example Extend™ or Arena™, the advantage of using a timed net is generality and flexibility. Arena™ for example is an excellent tool for simulation of queueing systems. Problems start when the system simulated does not fit this view. Programming user-defined blocks is time-consuming and not easy, especially if the user is not familiar with the host language (Siman in this case). Using timed nets has to be supported by user-friendly simulation environments with automatic collection of effectiveness parameters. PetriSim is an attempt in this direction. A disadvantage of using timed nets is the simplicity of the graphical language resulting in fast growth of the network size. This problem can be resolved by some kind of hierarchy, which together with an appropriate preemption mechanism will direct future development of PetriSim.

**REFERENCES**


OUTPUT ANALYSIS OF CYCLICAL SIMULATIONS

Rita Marques Brandão
Departamento de Matemática
Universidade dos Açores
Apartado 1422, 9501-801 Ponta Delgada
Portugal
E-mail: rita@notes.uac.pt

Acácio M. O. Porta Nova
Departamento de Engenharia e Gestão
Instituto Superior Técnico
Av. Rovisco Pais, 1049-001 Lisboa
Portugal
E-mail: apnova@ist.utl.pt

KEYWORDS
Discrete event simulation, output analysis, simulation meta-models, time series, SARIMA models.

ABSTRACT

In this paper, we present an approach for analyzing the output of non-terminating cyclical simulations. Deterministic components are estimated with nonlinear regression and Fourier series, while SARIMA models are used for the residuals. The approach is illustrated with a single lane traffic analysis. Some conclusions and suggestions for further work are stated.

INTRODUCTION

In virtually all simulation studies, the output analysis of discrete event simulation models is critically important. Indeed, the most perfectly crafted model can be rendered useless by a careless statistical analysis of the model-produced results. On the other hand, output analysis has been, as well, one of the most active research areas in simulation, for the last several decades. In fact, many innovative approaches have been proposed to extract useful information from the raw data generated by time-consuming computer programs. One such approach is the classical time series (Box et al. 1994) method. Initially proposed by (Fishman 1971), it did not receive much attention, except for (Schriber and Andrews 1984). In both cases, the context was the steady-state analysis of stationary simulations. Recently, interest in this method revived; see (Brandão and Porta Nova 2003a,b). In contrast to previous work, our research has been focused on the use of classical time series models for analyzing non-stationary simulations. But, simulations with periodic or cyclical behavior still remain a mostly unexplored topic in the specialized literature. One exception is a brief reference by (Law and Kelton 2000) on how the method of independent replications could be used to obtain point and confidence interval estimates for cyclical means. However, many real-life systems of undeniable practical interest actually present some kind of cyclical behavior, namely in areas like traffic engineering, banking, inventory management and manufac-
turing.

This paper is organized as follows. First, we discuss how non-terminating cyclical simulations can be analyzed using seasonal time series models. Then, we illustrate our approach, analyzing a traffic problem. Finally, we present some conclusions and suggestions for further work.

TIME SERIES FOR CYCLICAL SIMULATIONS

It is well known that the output produced by discrete event simulation models is highly autocorrelated. This has driven analysts to exploit this dependence structure through the use of classical time series models. These models were either used as a means of building confidence intervals for selected performance measures—(Fishman 1971) and (Schriber and Andrews 1984)—or as meta-models for representing the responses; see (Brandão and Porta Nova 2003a,b). In our previous work, we used ARIMA(p,d,q) models for analyzing non-stationary (terminating) simulations. In some situations, it was necessary to previously apply a variance-stabilizing transformation.

A special type of a non-stationary simulation occurs when the target system presents some periodic or cyclical behavior. For example, in traffic management, red and green lights alternate for each direction, and so, queue lengths and sojourn times will oscillate between some limits. Basing our analysis on simple averages will certainly be misleading... We propose building special-purpose meta-models for representing the expected evolution, in time, of the selected responses. Regression methods and Fourier series allow the estimation of the cyclic deterministic component. Removing this deterministic component may, or may not, eliminate the periodic behavior. Thus, we felt that it would be sensible to investigate the applicability of the seasonal ARIMA (SARIMA) models (Box, Jenkins and Reinsel 1994) to explain an eventual cyclic stochastic component.

Our approach consists of the steps that follow.

(i) Averaging the response time series across runs.

In fact, we are not restricted to a single realization of the
response time series, in contrast to what happens in econometrics, for instance. Thus, the underlying expected behavior of the response becomes more apparent, with a much smaller variability. We have found out that, in many situations, 30 replications is a good choice, from both the statistical and the experimental viewpoints; see (Brandão and Porta Nova 2003b).

(ii) Meta-model identification.

We analyze the estimated autocorrelation (ACF) and partial autocorrelation (PACF) functions, in order to identify suitable transformations for achieving a stationary time series. Sometimes, the appropriate transformation involves removing a trend or cyclical deterministic component, before applying the Box-Jenkins methodology. If the transformed response still presents a periodic or cyclical behavior, we have to identify the period (s) and determine the two orders of differencing (seasonal, D, and non-seasonal, d) for the candidate SARIMA models. Then, inspecting the estimated ACF and PACF of the transformed series, we select a candidate ARMA(p, q) × (P, Q)s model. The orders P and Q are chosen analyzing the estimated ACF and PACF at lags which are multiples of s. The orders p and q are chosen analyzing the estimated ACF and PACF at lags 1, 2, ..., s

(iii) Meta-model estimation and diagnostic checking.

It is advisable to estimate the meta-model using an automated procedure. Ours is taken from the R Language (Ihaka and Gentleman 1996). The diagnostic checking of the model is done applying the portmanteau lack-of-fit test (with the modified Ljung-Box-Pierce statistic) and checking for correlation between the estimated parameters. For choosing between several candidate models, we use Akaikes information criterion (AIC).

(iv) Using the fitted meta-model.

The fitted meta-model allows us to evaluate the target system and answer many “what-if” questions, without further simulation. For instance, when a road is partially closed for repairs, where do we place the traffic lights, in order to accommodate waiting cars? In a worst case scenario, how long must an incoming driver wait to pass the affected section? In a rush hour period, at what rate will the waiting line increase?

A TRAFFIC ILLUSTRATION

Our approach to cyclical non-terminating simulation analysis is illustrated using a slightly modified version of the Single Lane Traffic Analysis classical example; see (Nozari et al. 1984) or (Pritsker and OReilly 1999). In a two-lane road, with the traffic flowing from both directions, one of the lanes has been closed for 500 m, for repairs. At each end of the closed section, traffic lights were installed for controlling the traffic flow, allowing traffic (from only one direction) to flow for a specified time interval. When a light turns green, the waiting cars (from that direction) start moving and pass the light every two seconds. A car arriving when there are no waiting cars and the light is green, will pass through the light without delay. We assume that the time to pass through the repair zone is 60 sec. A light cycle consists of green in direction 1, both red, green in direction 2, both red, and then the cycle is repeated. Both lights remain red for 55 sec to allow the cars in transit to leave the section under repair, before traffic from the other direction is initiated. During most of the day, car inter-arrival times are exponentially distributed with an average of 12 sec from direction 1 and 9 sec from direction 2. However, by the end of the day, there is a one-hour rush period and the expected inter-arrival time from direction 2 is halved to 4.5 sec. In (Brandão and Porta Nova 2003b), we simulated green-light times at 5 sec intervals around 60 sec and concluded that the cycle that performed better was 55-50-55-60.

In this work, our objective is to investigate if it is feasible to build adequate meta-models for characterizing the cyclical behavior of the system, both under normal and congested conditions. Two responses of different nature will be analyzed: the average queue length and the average time spent in the system.

We performed 30 independent replications of the corresponding simulation models, with a reference duration of 3600 sec. The duration varied, because each run was only terminated when the last entity that had arrived before 3600 sec left the system. Initially, each model was started in a empty and idle state. Queue lengths (from one direction) were collected at regular 10 sec intervals, with \( t = 10, 20, \ldots, 3600 \), starting at the beginning of the red light for the opposite direction. Then, the corresponding observations across runs were averaged. Sojourn times were sorted according to the time interval in which the arrival had occurred. Then, for each time interval, we collected only the first observation and the corresponding observations across runs were averaged. Additional observations in the same time interval were rejected to maintain independence. This also contributes to a more stable behavior of the response. Then, we tried several transformations for estimating and removing the trend and/or the cyclical deterministic components, using Fourier series and the nonlinear least squares method. For removing the trend component, we fitted a straight line. The cyclical deterministic component was approximated using an asymmetric triangular wave, expressed as a Fourier series. Finally, we applied the Box-Jenkins methodology to the residuals of the deterministic fit of each response (it will be referred as the stochastic component), through identification, estimation and diagnostic checking of the SARIMA(p, d, q) × (P, D, Q)s models.

Applying our approach to the data corresponding to the normal conditions, we were able to fit an asymmetric triangular wave to the deterministic component and a
SARIMA\((1,0,0)\times(0,1,1)_{22}\) model to the stochastic component of the average queue length for directions 1 and 2. Although we performed the experimentation for both directions, we only present the graphs corresponding to direction 2, because the results are very similar.

In Figure 1, the original series (dash-line) and the corresponding fitted deterministic series (solid blue line) of direction 2 are represented.

![Average Queue Length, Direction 2](image)

**Figure 1: Original and Fitted Series: Average Queue Length, Direction 2**

Comparing the two curves, we see that both series have basically the same behavior, indicating that the fitted deterministic model properly represents the original data series. A more detailed analysis (see Figure 2) reveals insignificant discrepancies between the two curves.

The fitted meta-model allows us to answer one of the questions formulated at the end of the second Section. The maximum value for the average queue length for direction 1 is approximately 14 and 18.5 for the direction 2. Considering that an average car length is 4.5 meters and that, for instance, 0.5 meters should be left between adjacent cars, then, at least, 70 and 93 meters must be left before the crossroads for directions 1 and 2, respectively, in order to avoid traffic jams.

![Average Sojourn Time, Direction 2](image)

**Figure 3: Original and Fitted Series: Average Sojourn Time, Direction 2**

We performed a similar analysis for the other response, the average sojourn time. Again, it was possible to fit an asymmetric triangular wave to the deterministic components for the two directions. With respect to the stochastic components, we fitted a SARIMA\((3,0,0)\times(0,1,1)_{22}\) model for direction 1 and a SARIMA\((2,0,0)\times(0,1,2)_{22}\) model for direction 2.

Except for a few extremal values, the global behavior of the original series is mostly captured (see Figure 3).

![Average Queue Length Direction 2](image)

**Figure 2: Original and Fitted Series: Average Queue Length, Direction 2 (1500 to 2000 sec)**

Even a more detailed scrutiny (see Figure 4) only shows insignificant discrepancies.

Having this meta-model available, we can answer the second question we formulated earlier (iv). Under normal traffic conditions, the expected maximum sojourn time is approximately 235 and 222 for directions 1 and direction 2, respectively.
The one-hour rush period turns out to be a very interesting situation if the 55-50-55-60 cycle is left unchanged. Analyzing both system responses for direction 1 yielded similar results, although the fitted models for the stochastic components were distinct: a SARIMA(1, 0, 1) × (0, 1, 1)_{22} model for the average queue length and a SARIMA(2, 0, 0) × (0, 1, 1)_{22} model for the average sojourn time.

However, when we consider direction 2, both responses grow without bounds; see Figures 5 and 6, respectively. This is no surprise, since, in this case, the corresponding utilization factor is greater than 1. We can see that the average queue length shows a linear trend, with a cyclic component superimposed on it. On the other hand, only a linear trend is visible for the average sojourn time. The fitted models for the stochastic component were a SARIMA(0, 1, 0) × (0, 1, 1)_{22} model for the average queue length and an ARIMA(3, 1, 1) model for the average sojourn time. Again, the fitted deterministic components keenly capture the essential behavior of the original series.

These meta-models finally allow us to answer the last question in (iv). Thus, under congested traffic, the average queue length for direction 2 linearly increases at a rate of about 0.086/sec (5.16/min), with an asymmetric triangular wave of approximate amplitude of 21.5 superimposed on it. The average sojourn time basically follows a straight line, with a slope of 0.63 and an intercept of 145 seconds, approximately.

CONCLUSIONS

Our recent work has been focused on extending the use of classical time series models to non-stationary simulations. In this paper, we present a very effective approach for analy-
zizing cyclic behavior in non-terminating discrete event simulations. When applied to a classical example from the simulation literature, our approach was capable of producing reliable results of undeniable practical interest.

However, we feel that it is still possible to improve the approach and extend it to other classes of terminating or non-terminating simulations. Also, more extensive experimentations may be performed, and many other analytical results investigated...

REFERENCES


AUTHOR BIOGRAPHIES

**RITA M. BRANDÃO** is a Lecturer and Ph. D. student of the Department of Mathematics at the University of Azores. Her e-mail address is <rita@notes.uac.pt>.

**ACÁCIO M. O. PORTA NOVA** is an Associate Professor of the Department of Engineering and Management at the Superior Technical Institute (IST) of the Technical University of Lisbon. He is a member of IIE. His e-mail address is <apnova@ist.utl.pt>.
A SPARSE ORTHOGONAL FACTORIZATION TECHNIQUE FOR CERTAIN STIFF SYSTEMS OF LINEAR ODES

Pablo Guerrero-García and Ángel Santos-Palomo
Department of Applied Mathematics, University of Málaga
Complejo Tecnológico, Campus Teatinos
29071 Málaga (Spain)
E-mails: {pabitlo,santos}@ctima.uma.es

KEYWORDS
Sparse orthogonalization, Givens rotations, stiff systems of linear ODEs, implicit integration methods.

ABSTRACT
The discretization of a stiff system of linear ODEs leads to a sequence of systems of linear equations when implicit numerical methods are used. The dynamical behaviours considered in this paper are those modeled by large systems of $n$ ODEs in which the discretized Jacobian $J(t_k)$ at step $k$ of the simulation is a positive semidefinite symmetric matrix of the form $A_k A_k^T$, where $A_k$ is a subset of the columns of a sparse matrix $A \in \mathbb{R}^{n \times m}$ ($m \gg n$) with a fixed sparsity pattern, and $A_{k+1}$ is obtained by appending/deleting columns to/from $A_k$. A sequence of systems with coefficient matrix $\sigma I_n + A_k A_k^T$ (constant $\sigma$) has to be solved, and its sparse Cholesky factorization is maintained into a static data structure to take advantage of the fact that each system in the sequence is closely related with the previous one. The sparse approach has the added bonus of being parallelizable, and row orderings can be exploited to decrease the computational effort needed when a refactorization has to be done.

1. THE STIFF SYSTEM OF LINEAR ODEs TO BE INTEGRATED
The dynamical behaviours considered in this paper are those modeled by large systems of $n$ linear ODEs

$$y'(t) = J(t) \cdot y(t) + b(t), \quad \alpha \leq t \leq b,$$

$$y(t_0) = \alpha, \quad y(t) \in \mathbb{R}^n,$$

where $J(t) \in \mathbb{R}^{n \times n}$ is the Jacobian matrix. When this system of ODEs is stiff, its integration is usually carried out by using an implicit numerical method (Stuart and Humphries, 1999), where we shall denote $y(t_k) \approx y_k$ and $y'(t_k) \approx y'_k$.

Furthermore, we are interested in cases in which an explicit formula for $J(t)$ is not known, but instead the discretized Jacobian $J(t_k)$ at step $k$ of the simulation is known to be a positive semidefinite symmetric matrix of the form $A_k A_k^T$, where $A_k$ is a subset of the columns of a sparse matrix $A \in \mathbb{R}^{n \times m}$ ($m \gg n$) with a fixed sparsity pattern, and $A_{k+1}$ is obtained by appending/deleting columns to/from $A_k$.

Let us show that all we have to do is to solve a sequence of linear systems of equations. For example, if a backward differentiation formula

$$y_k = \sum_{j=1}^{m} a_j y_{k-j} + h \cdot b_0 y'_k$$

is used, then one system of linear equations

$$(\sigma I_n + J(t_k)) y_k = \sigma \sum_{j=1}^{m} a_j y_{k-j} - b(t_k), \quad \sigma \doteq -(hb_0)^{-1},$$

must be solved at each step $k$ to compute $y_k$; on the other hand, using a $q$-stage diagonally implicit Runge-Kutta method

$$y_k = y_{k-1} + \sum_{s=1}^{q} c_s K_s, \quad \sigma \doteq -(hb_{ss})^{-1},$$

where $f(t, y(t)) = y'(t)$ and $b_{ss}$ is constant for all $s \in 1:q$, then $q$ systems of linear equations

$$(\sigma I_n + J(t_{k-1} + a_s h)) \cdot K_s = \sigma h \cdot \varphi(t_{k-1} + a_s h), \quad (s \in 1:q),$$

must be solved at each step $k$ to compute $y_k$, where

$$\varphi(t_{k-1} + a_s h) \doteq J(t_{k-1} + a_s h) \cdot y_{k-1} + \sum_{j=1}^{s-1} b_{sj} K_j + b(t_{k-1} + a_s h)$$

and the same assumptions about the computability of $J(t_k)$ also hold for $J(t_{k-1} + a_s h)$.

Another sparse proposal to deal with a similar sparse problem was described by Zlatev (1991, §8). There are strong differences with our sparse approach, namely:
1. He assumed that $J(t)$ is a general sparse matrix, hence both the special property and pattern of our $J(t)$ cannot be taken into account in his proposal. In fact, he remarked that this exploitation could lead to more efficient approaches.

2. He did not use a static structure, but an incomplete sparse LU factorization of the discretized Jacobian was recomputed from scratch every step of the simulation, with an automatic adjustment of the drop-tolerance for this factorization. Therefore, he cannot take advantage of the fact that each system in the sequence is closely related with the previous one, which is exploited in our approach by updating a sparse orthogonal factorization.

3. Our sparse approach has the added bonus of being parallelizable, and row orderings can be exploited to decrease the computational effort needed when a refactorization has to be done because of round-off error accumulation.

The rest of the paper is organized as follows. In §2 we describe our sparse proposal, and some parallelizability and refactorization issues are treated in §3. Finally, some conclusions are included.

### 2. A SPARSE ORTHOGONAL FACTORIZATION TECHNIQUE

Our original sparse orthogonal approach uses the sparse QR factorization of $A_k^T$ to maintain a sparse Cholesky factor $R_k$ of $A_k A_k^T$, with $A_k$ being a full column rank matrix (Santos-Palomino and Guerrero-García, 2002). Adding and deleting rows to $A_k^T$ is allowed, which is respectively known as row updating and downdating the factorization. Since $A_k$ is formed by a subset of the columns of a fixed matrix $A$, we adapt Saunders’ techniques for square matrices (Saunders, 1972) to matrices with more rows than columns, using the static data structure of George and Heath (1980) but allowing row downdating on it. We work on top of the static structure of the Cholesky factor $R$ of $A A^T$, and a permutation of the rows of $A$ takes place a priori to improve the sparsity of $R$. A well-known result from sparse linear algebra is that $R_k \subset R$, namely that we have enough memory already allocated in the static structure which has been a priori set up whatever $A_k$ (and consequently $R_k$) is. The interested reader can find the details in (Santos-Palomino and Guerrero-García, 2002), and a MATLAB toolbox based on this technique has been developed in (Guerrero-García, 2002).

In §1 we have shown that the problem we have to solve is reduced to an efficient way to deal with a sequence of sparse linear system of equations whose coefficient matrix is $\sigma I_n + A_k A_k^T$ (constant $\sigma$), which is positive definite. The solution of such systems can be done in $O(n^2)$ operations in the worst case by a forward and a backward triangular substitution once the Cholesky factor $R_k$ of its coefficient matrix is available. Firstly, we state the following simple but crucial fact.

**Theorem 1** The Cholesky factor of $\sigma I_n + A_k A_k^T$ coincides (barring signs) with the triangular factor of the QR factorization of $[A_k, \sqrt{\sigma} I_n]^T$, namely

$$\sigma I_n + A_k A_k^T = R_k^T R_k \iff \begin{bmatrix} A_k^T \\ \sqrt{\sigma} I_n \end{bmatrix} = Q_k \begin{bmatrix} R_k \\ O_k \end{bmatrix}.$$  

**Proof** From elementary matrix algebra, applying the orthogonality of $Q_k$:

$$R_k^T R_k = \begin{bmatrix} R_k^T, O_k \end{bmatrix} Q_k^T Q_k \begin{bmatrix} R_k \\ O_k \end{bmatrix} =$$

$$= \begin{bmatrix} A_k, \sqrt{\sigma} I_n \end{bmatrix} \begin{bmatrix} A_k^T \\ \sqrt{\sigma} I_n \end{bmatrix} = \sigma I_n + A_k A_k^T.$$  

The presence of the Marquardt-like term $\sigma I_n$ in the expression $\sigma I_n + A_k A_k^T$ to be factorized allows us to dispense with the full column rank requisite in our original approach; moreover, we can also deal with cases in which $A_k$ has more columns than rows. From Theorem 1 above, all we have to do is an initialization from $R = \sqrt{\sigma} I_n$ in the diagonal of the static structure (note that obviously this diagonal elements do not destroy the predicted static structure). Then, each row of $A_k^T$ will be linearly dependent of the rows already present in the structure, hence a sequence of Givens rotations has to be performed to annihilate all nonzero elements (and eventual fill-ins) of each row of $A_k^T$. Since all rows of $\sqrt{\sigma} I_n$ remains into the structure during the whole process, downdating is done as in our original approach. An illustrative example can be found in the Appendix.

A FORTRAN implementation of the updating process (without the downdating) can be found in the SPARSPAK package developed by (George and Liu, 1981), which is currently being translated into C++ as described in George and Liu (1999). The availability of this SPARSPAK++ package implies a rapid prototyping of a low-level implementation of the sparse proposal given here.

### 3. PARALLELIZABILITY AND REFACTORIZATIONS

In this section we are firstly going to deal with some issues about the parallelizability of the sparse technique described in §2.

The papers of Bendtsen et al. (1995) and Kratzer (1992) are closely related to our sparse approach, both working
with SIMD architectures. Bendtsen et al. (1995) parallelize the row updating and downdating processes in a dense QR factorization of $A^T_k$ under the assumptions that this matrix has more rows than columns and full column rank, and that no change in range occurs after the update or the downdate; furthermore, they consider no static structure in spite of the fact that they dispense with the orthonormal factor. On the other hand, Kratzer (1992) only parallelizes the updating process in the computation of a sparse row-sequential QR factorization on top of the static structure of George and Heath (1980) using a data-driven approach. Note that in both parallel proposals the presence of the $\sigma I_n$ term does not invalidate the results, and then a sparse parallel implementation along the lines of Kratzer’s work is feasible, once extended to deal with downdatings.

As commented by Björck (1996, §6), a decrease of the computational effort when a refactorization has to be done can be accomplished by taking into account a suitable row ordering of $A^T$. The key point is that the column order of $A_k$ does not affect the density of the Cholesky factor $R_k$ of $\sigma I_n + A_k A_k^T$. Our MATLAB implementation takes advantage of this fact by using the sparse option of the qr command, which is an executable file (originally written in C, no source code available) implementing the updating process (without the downdating) referred to in §2.

4. CONCLUSIONS

A sparse orthogonal factorization technique for the integration of certain stiff systems of linear ODEs has been proposed. The dynamical behaviours modeled by this kind of systems are not as general as those addressed by Zlatev (1991), but this is just the reason why we can take advantage of several sparsity features such as maintainability of the factorization, management of a static data structure, parallelizability and a decrease of the computational effort when a refactorization has to be done.

REFERENCES


APPENDIX: AN ILLUSTRATIVE EXAMPLE

To get some insight we consider (1) with $m = 2$, $a_1 = 4/3$, $a_2 = -1/3$ and $b_0 = 2/3$. If $t_0 = b$, then $h = (a - b)/N < 0$ and hence $\sigma > 0$ (e.g., $N = 15000, a = 0$ and $b = 1$ implies $\sigma = 10^4$). We have borrowed $A \in \mathbb{R}^{27 \times 51}$ from AFIRO, a well-known benchmark linear program from the NETLIB collection. In each one of the $N$ steps a random choice of how many adding and dropping columns is done, from among the 51 columns of $A$. The initialization was done from $R = 100 \cdot I_{27}$, and a refactorization was triggered when $\|\sigma I_n + A_k A_k^T - R_k^T R_k\|_\infty > \varepsilon$, where $\varepsilon \approx 2.22 \cdot 10^{-16}$ is our machine epsilon.

AUTHOR BIOGRAPHY

PABLO GUERRERO-GARCÍA received his MSc in Computing from the University of Málaga in 1994, his MSc in Mathematics from the Complutense University of Madrid in 1998, and his PhD in Mathematics from the University of Málaga in 2002. He is a teaching assistant in Numerical Analysis at the University of Málaga from 1994.

ÁNGEL SANTOS-PALOMO received his MSc in Mathematics from the University of Sevilla in 1975, and his PhD in Econometrics from the University of Málaga in 1995. He was a teaching assistant in Numerical Analysis at the University of Málaga from 1975, where he became Professor.
FORMALIZED WORLD VIEWINGS - A WAY TO NESTED SIMULATION

Eugene Kindler
Department of Mathematics
University of Ostrava, Faculty of Sciences
CZ – 701 03 Ostrava, Dvorakova 7,
Czech Republic
E-mail: Kindler@ksi.mff.cuni.cz

Ivan Krivý
Department of Computer Science
University of Ostrava, 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 Cèzeaux, F – 63177 Aubière, France
E-mail: tanguy@isma.fr

KEYWORDS
Object-oriented simulation, Reflective simulation,
Production/logistic systems

ABSTRACT

The paper deals with main classes, i.e. object-oriented programming classes containing definitions of other classes. Main classes represent world viewings and can be widely applied in simulation. Experiences are presented and the occurrence of structures of different world viewings in a model is made clear. The relation between the main classes and model nesting is explained.

1 INTRODUCTION

There are thousands of such simulation studies. They are usually classified according to the simulated systems but sometimes another classification can occur – according to applied simulation tool, according to experimental frames etc. In the sections 1 to 4, let us concentrate to the classification according to the simulated systems.

The main problem in simulation is to construct a good model in a short time. Under the words good model, it is to understand that the model corresponds to the simulated system. Simulation languages tried to solve this problem: they were designed and implemented with the aim to offer the simulationist a language by which he would be able to describe the simulated systems (and not the model) so that the description would be automatically transformed into one corresponding simulation model running on a computer. It is evident that the simulation languages have to be similar to professional languages used in various domains of science, technology and social management. Moreover, it is obvious that the larger the set of the simulation domains becomes, the larger the set of simulation languages is needed. The performances and memory capacity of the computers implied increasing demands for reflecting more or less details of the simulated objects in their computer models – in exact way, we are to speak that the increasing computer capacities cause that more detailed systems are defined in using simulated objects and these systems are simulated.

Already in 1966, the growth of the spectrum of simulated systems and their details has stimulated ideas (Dahl and Nygaard 1967; 1968; Garvick 1968). After many years of their existence, a certain choice of ideas presented in (Dahl and Nygaard 1967; 1968) has been widely applied since the eighties and called object-oriented programming (in the further text shortly OOP). The principle of OOP was that new programming languages could be implemented by defining classes so that classes were encapsulating – among others – methods that could be applied as a new language construction. For example, a message (demanding object $S$ to perform method $M$ with a parameter $P$) could represent a phrase in which $S$ was a subject, $M$ a verb and $P$ an object or complement, or in which $S$ and $P$ were objects and $M$ a conjunction or a preposition: examples based on (Dahl and Nygaard 1968) are $S$.enter($P$) and $S$.into($P$) in list processing.

Nevertheless, in (Dahl and Nygaard 1967) and (Dahl and Nygaard 1968) more ideas were presented than accepted in OOP. For the purpose of the present paper, the idea of main class is important. Main class is a class containing definitions of other classes. While a conventional class applied in OOP is a representation of a concept the main class is a representation of a world view. If definitions of classes $A, B$ etc. occur in a main class it represents a world viewing using concepts represented by $A, B$ etc.

A main class can be used as any other classes, i.e. it can have parameters/attributes and methods and it can be specialized to its subclasses. For example, the main class geometry, containing classes like point, circle, line, square etc. can have attributes representing a coordinate system and methods like those for generating the coordinate system, testing whether two lines are parallel etc., and can be specialized e.g. to classes representing mechanics, view to colored configurations, view to geographical maps etc. If $D$ is a specialization of a main class $C$ containing definitions of classes $A, B$ etc., then $A, B$ and other such classes can be
used inside $D$ and can be specialized, too. For example point can be specialized to mass point (adding attribute mass), colored point (adding attribute color) or to cartographic symbol.

The main class arises by a certain domain analysis (Hill 1996), i.e. by analyzing a certain domain of science, technology or world viewing that is not yet considered as an official science. The main class is a certain formalization of such a domain. One can consider a main class also as a representation of certain meaning ability – for example the class geometry mentioned above can be considered as class geometry. Using such a class, we get a computer model of an expert working in the given domain. Let a specialization $D$ of a main class $C$ adds certain set $G$ of concepts to the contents of $C$; then $G$ corresponds to a certain enriching of the world viewing or of the human carrier of $D$. The latter behaves like knowing more about the domain than the human carrier of $C$ but can effectively apply it only in a subdomain of the domain handled by $C$.

2 EXPERIENCES WITH WORLD VIEWINGS

The block-orientation of the OOP language Simula (Dahl and Nygaard 1967; 1968; Dahl et al. 1968; Simula Standard 1989) enables to define the main classes. One of them is class Simulation containing definition of class process. This main class is a standard class of Simula. It represents a viewing of a world with Newtonian time and with ordered lists, while class process represents a general concept of an object which exists inside such a world and can interact with the other objects of the same type. In the early seventies of the XX century, the users applied class Simulation to specialize it to several main classes like those for combined systems and several main classes admitting to program in Simula like in the old and popular language GPSS.

Nevertheless, it was possible to conceive subclasses that are more specialized. Sometimes the specialization was iterated. Let us present some examples (instead of class Simulation we use the abbreviation $S$):

2.1. $S$ was specialized to reflect a domain of farm works with transport by cars (Kindler et al., 1983). This main class was applied to models of crop works and sowing. With the advent of a modern manner of transport, the main class was specialized to include containers and their carriers and more complex systems were simulated (e.g. pork production, veterinary sanitation).

2.2. $S$ was specialized to class GIW reflecting general viewing to iron works – furnaces, cranes and other transport tools etc. GIW was then specialized to reflect modern technologies of steel production (continuous casting). Both the classes were then specialized according to viewing different iron works existing in the Czechoslovak Republic: the viewing covered the individual differenteness one establishment from the others. The specialized main classes enabled the modelers to describe the simulation models “almost in the languages used in the corresponding steel works by their managers”. Some main classes were then specialized according to the level of abstraction.

2.3. $S$ was specialized to class GMP reflecting general viewing machine production centers – NC-machines and other processing engines, stores, conveyors, production programs etc. The main classes were then specialized similarly as in 2.2 according to rules, configurations (and abstraction levels, too) existing at different Czechoslovak machine works.

2.4. $S$ was specialized to class GFT reflecting a general viewing of fault trees, i.e. systems composed of generators of Boolean values (in fault / not in fault) and of blocks performing Boolean OR and AND operations, connected by channels which rarely compose cycles. “Macros” composed of such elements were admitted. GFT was then specialized according to various views of nuclear power plants.

2.5. $S$ was specialized to class GLP reflecting general views of systems with local propagation (Krivy and Kindler 1997). It was then specialized to reflect the properties of epidemics, forest fires and to inspect pests. Some of them were further specialized according to knowledge of their propagation.

Other application of main classes forming concerned compartmental models, nuclear hygiene and transport of radioactive wastes.

3 PARALLEL WORK OF SIMULATION MODELS

A habitual simulation study is a sequence of ordered simulation experiments so that two simulation experiments cannot mutually overlap. A traditional view is that a simulation experiment is a stable part of the simulation program surrounded and governed by the experimental frame, which can activate the simulation experiment as a new run of the simulation model. Simula offers an efficient tool for such a purpose; it is called prefixed block. It has a form PREFIX begin BLOCK CONTENTS end, where PREFIX is any class and BLOCK CONTENTS describes what the block should do. In this description, one can use any tool defined in the PREFIX. In practice, PREFIX is a main class and in case of simulation BLOCK CONTENTS is a description of the model used in the simulation experiment. Inside it, one may use any tools introduced in the main class PREFIX. This practice leads to an impression that such a prefixed block is a stable component of the simulation program.

Nevertheless, Simula allows in principle dynamic allocation of blocks, including prefixed ones. The occurrence of dynamic allocation of block has been known since the 60-ies of the XX century (Naur 1960) e.g. in case a block $b$ is a part of a body of a program $f$ so that $f$ is recursively called from inside $b$. If a block $b$ has an entity $e$, then any allocation of $b$ has its own $e$ which can be a variable, a procedure but a class as well. In case $b$ corresponds to a simulation experiment, then several contemporaneous allocations of $b$ correspond to several contemporaneous simulation experiments.
This possibility was applied in (Weinberger 1987) for a continuous optimizing during progress of several simulation models running in parallel. A simulation study was a model of a session of several simulationists who try to get optimal parameters of a certain system \( S \). Each of them has its proper computer simulating \( S \) with parameters according to the opinion of its owner. The owners mutually exchange information on the behavior of their models, so any simulationist can learn from the experiences of his colleagues and change his own opinion. In this case, the block corresponding to the simulation experiment was allocated \( n \)-times where \( n \) was the number of the simulationists. Note that it is the same block that is several times allocated.

4 NESTED WORLD VIEWINGS

Simula allows nesting blocks and classes. A statement of a block or of an entity (a class or a subprogram) can be a block and among the declarations of a block or a class declaration can occur. Such a class declaration can be this of a main class. That allows possibilities with rather fantastic horizons but already nowadays such possibilities can be applied in very realistic cases.

Let \( C \) be a main class (world viewing) that contains a declaration of class (concept existing in world viewing \( C \)) called \( H \). Let the declaration of class \( H \) contains a procedure (a method) \( F \). Then any instance \( E \) of class \( H \) is able to perform \( F \). The rough form of the declaration of \( F \) is an algorithm, i.e. a sequence of statements. Any of them can be a block prefixed by a main class \( D \). When \( E \) is demanded to perform \( F \) the computing can enter into such a block. This phenomenon can be metaphorically interpreted so that the element \( E \) enters into a certain phase of its existence, in which it is possible to “think”, using the concepts introduced in \( D \), or – in other words – that \( E \) enters into a certain phase of its existence, in which it actively applies the world viewing \( D \).

With a less sense for a metaphor, we could say that in this phase of its existence, \( E \) models something that could be understood if the world viewing \( D \) is applied. The modeling phase performed by \( E \) can be a simulation modeling one or not. The simulation modeling phase comes when \( D \) can be applied similarly as mentioned in the preceding sections. This special case is discussed in section 5 of the present paper. Now, let us concentrate to general properties of the case of nested world viewings.

OOP is also widely applied outside simulation and, therefore, a use of world viewing as well. The example of geometry – shortly mentioned in section 1 – is a demonstration of that proposition. A wittily declared main class can effectively help in programming – instead of a tiresome algorithmization of a process that should be computerized, i.e. instead of transforming the results of a mental analysis into a “monoprocessor” program. One can map the ideas the modeller has in his mind during the analysis into classes and cover them by a certain main class. It is a true image of the real world viewing that the programmer used during the analysis of the matter to program.

Beside the mentioned world viewing of a specialist in geometry, there are of examples applied by the authors or by their colleagues and university students. Let us mention some illustrations.

4.1. In (Dahl 1966) a world viewing was presented to show how to compute the shortest path. Pulses disseminate from the start place in all possible ways that come out from the start point. When a pulse enters a branching \( B \), it multiplies so that new pulses come from \( B \) along all ways leading out from \( B \), and \( B \) is marked in a sense that “it was already visited”. What was just stated comes into being only when a pulse enters a branching that was not marked; otherwise the pulse disappears without any continuation. Every pulse carries information on the pulse that generated it and on the way that he went along. When a pulse meets the target it itself and its “genealogy” inform on the short path – therefore, when such an event occurs, it is not necessary to continue viewing on the pulse moves and multiplying. In (Dahl 1966) that world viewing was formalized as a certain simulation model of the fantastic world of the pulses.

4.2. In a special problem concerning a computer model of a rectification column, it was necessary to implement a certain new numerical method for solving partial differential equations; the method admonished of about predictor-corrector methods of applied in ODE solvers, but the partial differential equations needed to implement a certain iterated “dialogue” among the predictors who computed their values as spline extrapolations from various space-time directions. Therefore, the predictors were formulated as certain “micromathematicians” existing in the rectification column and playing a certain game in anticipating the future states of the system elements. The viewing on the world of the micromathematicians was different from the world viewing to the cascade process in which it was applied. The method was shortly explained in (Kindler and Weinberger 1992).

4.3. Several input files had to be processed. The records of them has form \( <t, a, b, c> \), where \( t \) represented time when \( a, b \) and \( c \) were obtained. The records of every time were ordered according to \( t \) but the values of \( t \) were not synchronized: for a value of \( t \) existing in a record of a file, no records with the same value of \( t \) were guaranteed in the other files. The values \( a, b \) and \( c \) of all the files concerned the same process and the task was to get a unique file \( F \) that would inform on the process, i.e. to range the records with isolated values of \( t \) into \( F \) and to compute the average values \( \alpha, \beta \) and \( \gamma \) of \( a, b \) and \( c \) occurring in records with the same value of \( t \) and to range record \( <t, \alpha, \beta, \gamma> \) into \( F \). Similarly as in case of 4.1, the routine performing the described task was implemented as a simulation model of a fictitious system that existed in time occurring at \( t \) the files. It read the records in the appropriate times and handles them immediately, ranging them into \( F \).
A lot of other world viewing arose in everyday programming work for the implementation of more complex products. However, let us advert to the relation concerning simulation.

The world viewings mentioned in 4.1 and 4.2 were effectively applied in implementing special routines of simulation models, namely of that rectification column in case 4.2 and in many times in case 4.1 (e.g. in models of a container terminal (Kindler 2000) or of a production system (Kindler and Brejcha, 1990)). So one got a nesting model inside a simulation model, or – in other words – a main class of a special world viewing carried by an instance of a concept existing in another world viewing.

The world viewing 4.2 was applied in the concluding phase of a simulation study (a set of simulation experiments) S for processing a set of protocols generated during every simulation experiment of S.

In general, one can say that in a conventional simulation there are two levels – this is of the simulation experiment and that of the simulation study. While for an experiment the ascending time flow is typical, the study may be characterized as a certain model of a system of “eternal creating”. In this system, there is no time but different worlds (with their own Newtonian times) can be created, observed and possibly liquidated. Also this system could be described by viewing its proper “world”. As a simple introduction of such an application, we can present the viewing on a world with “eternal uncertainty”.

Simula has elementary tools for generating pseudorandom numbers. For example, function \texttt{normal}(a, b, U) generates random numbers of normal distribution with mean value \( a \) and standard deviation \( b \); \( U \) represents the “seed” (stream). Although these tools are very suitable from the numerical point of view, they can be applied as base in the OOP to form more “tangible” concepts, namely those representing automata having their own sort of distribution (including the parameters and the stream) and a certain “button” \( B \) so that their reaction, when pushing on \( B \), is to give a pseudorandom number.

Such automata could be viewed as certain elements of the simulated system, i.e. as something that corresponds to the concept used in the viewing of the world represented by the simulated systems. In other words, the classes of such automata could be considered as those declared in the main class applied to describe the simulated system. Nevertheless, we have got an experience to view such generators as the objects existing at the level of the “eternal creating”, as they represent something uncertain for us. The necessary generators are created once for the whole simulation study and are used by all its simulation experiments. The world viewing levels can be illustrated in Fig. 1, where the world viewing (main class) representations have rounded edges while their concepts (classes internal relating to the world viewings) are represented by rectangles.

\section{5 NESTED SIMULATION MODELS}

The nesting of auxiliary simulation models mentioned in 4.1 in a “real” simulation model of a container yard or in this of a production system leads to nested simulation. It is a simulation of a system that is represented as having one or more simulating elements. These elements can be fictive elements (like in the examples just mentioned) or humans who imagine some process evolving along a certain flow of time, or computers. The last case seems being the most important. Theoretically, a computer supposed in such a simulated system can simulate anything, but a very serious and relevant case occurs when the computer simulates its own environment. In such a case the whole simulation experiment (called external one) is described by a block \( B \) prefixed by some main class \( M \) while the experiment carried by a computer (called internal one) is described by a block \( b \) prefixed by class \( m \), which is similar to \( M \). In this case we speak on reflective simulation. When \( B \) is already allocated, it calls the prefixed block \( b \). Such models were implemented in the following cases:

5.1. simulation of a queuing system with a dispatcher anticipating possible consequences of his decisions by using a simulating computer (Kindler 1996),

5.2. simulation of a container terminal anticipating possible conflicts among the automatically guided vehicles by using a simulating computer (Kindler 2000),

5.3. simulation of a conveyor with rollers with several anticipations served by simulating computers (decision on continuing after a fault, decision on the optimal places of elaboration of the coming objects, decision whether a coming object should enter the conveyor immediately etc. (Berruet et al. 2003)),

5.4. simulation of a system of urban public bus transport, where the imagining of the passengers (occurring during

![Fig. 1: Scheme of Nesting](image-url)
their planning and attempting to find which sequence of busses is optimal to use) was modeled by simulation (Bulava 2002).

Note that in case 5.1 Simula allowed simulating a system, in which more than one dispatchers exist, they have their own simulating computers and decide more or less independently one from the others. Case 5.4 was an example where the simulating element is not supposed to be a computer but an imagining human. Notwithstanding, the model was implemented with a certain idea what would happen if the passengers were assisted by a center with a simulating computer that would replace the imperfect human imagining and recommend the people their optimal transport configuration, using telecommunication means.

5 CONCLUSIONS

Let \( M \) be a main class corresponding to a world viewing \( W \) with some classes-concepts \( A, B \) etc. At the present time the authors prepare software that would translate such a conventional main class \( M \) to another one \( m \), which world viewing would be enriched by a simulationist who is able to “look about” and to transform the observed situation into a simulation model to anticipate his future (Krivy et al. 2002).

The main class can be used also for other levels of a simulation study, namely for the experimental frame. Nevertheless, the nesting of simulation models formulates a question about the identity of the experimental frame: surely the external model should carry the experimental frame for the internal one.

In case 5.3 one applies several main classes: this for the communication between the computer and the modeler, that for the simulation study control, that for the description of the external model and three classes for the description of the internal models (different purposes of the internal models applications need similar but yet different world viewings).

REFERENCES


EUGENE KINDLER is Associate Professor at the University of Ostrava (Czech Republic) where he takes courses, seminars and research work in computer simulation and object-oriented programming.

IVAN KRIVÝ works at the University of Ostrava as Professor in applied mathematics. His main research activities are oriented to computer simulation and stochastic algorithms in solving global optimisation tasks.

ALAIN TANGUY is a researcher of LIMOS laboratory at the Blaise Pascal University in Clermont-Ferrand. His main research interests include object-oriented modeling and evaluation of manufacturing and transportation systems.
DRAWING PROCESSES SIMULATION: COMPARATION BETWEEN UPPER BOUND AND FINITE ELEMENT METHOD SOLUTIONS

Eva M. Rubio
Cristina González
Ana M. Camacho
Rosario Domingo
National Distance University of Spain (UNED)
C/ Juan del Rosal, 12
28040-Madrid
Spain
E-mail: erubio@ind.uned.es

KEYWORDS
Plate drawing, Plane strain, Coulomb friction, Upper Bound Theorem, Finite Element Method.

ABSTRACT
In the present work, plate drawing process has been studied by means of two different method such as Upper Bound Theorem (UBT) and Finite Element Method (FEM). The same process conditions (Coulomb friction and plane strain), geometry and technological parameters (die semiangle, reduction in cross-sectional area and Coulomb friction coefficient) have been used in the two cases. The deformation zone has been modelled in the UBT case by an only triangular rigid zone and in the FEM one by a mesh generated by the ABAQUS code used in this analysis (Hibbitt et al. 2003).

INTRODUCTION
Drawing is a metalforming process for obtaining of wires, rods, bars and plates which have a very wide application field in industrial sectors such as automotion, electric and others. Materials traditionally used in this type of manufacturing processes are aluminium and copper alloys and steels.

Basically, the process consist of reducing or changing the cross-section of this kind of pieces making pass them through a die by a pulling force. The main variables in the process are: die semiangle, α, cross sectional area reduction, r and friction coefficient, μf for representing the friction along the die-workpiece interfaces (Rowe 1986).

In general, to select the values of those parameters is not an easy task. Different methods to analyze the best combination of them have been developed (Rubio et al. 2003a). Among the methods more commonly used to analyze and simulate these processes are, for example, homogeneous deformation, slip lines field, slab method, upper bound theorem and finite element method (Rubio et al. 2003b).

The Upper Bound Theorem is one of the more interesting among the analytical methods because it allows:

- To provide an upper level of the power necessary to perform a certain process.
- To discriminate different components of the power.
- To calculate the friction in technological terms.
- To optimize the geometry of the selected model according to the kinematic criteria.
- To model the friction by different ways (partial friction and Coulomb).

Besides, it is easy to apply and does not require big resources of calculation. The main disadvantage of this method is, basically, the fact it only can be applicable under plane strain conditions.

As numerical method more widely spread in the last years, it is possible to mention the Finite Elment Method. It allows:

- To model the process taking into account a great amount of variables involved in it.
- To obtain multitude of very accuracy output variables.
- To compare results in a very intuitive way by simple observation of the simulations.

However, FEM needs bigger calculation resources and requires users with a good knowledge of the used software programme in order to achieve suitable results.

In this work, Upper Bound Theorem and Finite Elment Method have been applied to calculate the drawing force necessary in a plate drawing process carried out under plane strain and Coulomb friction conditions. The analysis has been made taking into account different technological parameter values which have already shown.

PROCESS ANALYSIS
As output variable to compare the results obtained by Finite Element Method and by Upper Bound Theorem, the adimensional drawing stress at the die exit, σf/2k, has been taken. This is very useful since it can be assimilated to the energy necessary to carry out the process by no dimensional terms.
Upper Bound Theorem

In the UBT case, the deformation zone has been modelled by an only triangular rigid zone (figure 1) because, in spite of other suitable models exist, this one fits better to the results found on the classical references about these themes (Hill and Tupper 1948). The election is based on the simplicity of this model and on its robustness versus variations of point B positions.

\[
\dot{W} = 2w\left[k\overline{AB}\Delta v_{i,2} + k\overline{BC}\Delta v_{j,2} + \mu p \overline{AC}v_2 \right]
\]

(1)

Where \( p \), pressure in the die, comes given by:

\[
p = \frac{\sigma_y}{AC(\text{sen} \alpha + \mu \text{cos} \alpha)}
\]

(2)

Expression obtained by forces equilibrium on the die and taking into account the friction. In the other hand, the necessary power to carry out the drawing process can be expressed by:

\[
\dot{W} = \sigma_y \dot{w}_j v_j
\]

(3)

Thus:

\[
\frac{\sigma_y}{2k} = \frac{\overline{AB}\Delta v_{i,2} + \overline{BC}\Delta v_{j,2}}{h_j v_2 - \frac{\mu}{\text{sen} \alpha + \mu \text{cos} \alpha}v_2}
\]

(4)

Finite Element Method

In the FEM case, the simulation has been carried out by means of the ABAQUS finite element code. The type of element that has been used in this analysis for meshing the workpiece is CPE4R (see figure 2). This is a continuum, 4-node, reduced-integration, plane strain, and hourglass control element.

Figure 1: Triangular rigid one zone model

To choose this element type the following aspects has been considered. The continuum (or solid) elements in ABAQUS can be used for complex nonlinear analyses involving contact, plasticity, and large deformations. A good mesh of quadrilateral elements usually provides a solution of equivalent accuracy at less cost than triangular ones. Besides, quadrilaterals have a better convergence rate than triangles, and sensitivity to mesh orientation in regular meshes is not an issue. In fact, ABAQUS recommendations for element usage are that if your problem involves contact or large distortions (as in the case of metalforming processes), you should use first-order quadrilateral or hexahedral elements. In addition, if the mesh distortion is severe, use reduced-integration ones.

To obtain \( \sigma_y/2k \) it is necessary to calculate the drawing force, \( F_j \). This variable can be calculated by means of NFORCE, an output variable of ABAQUS code. This nodal variable is obtained from the programme by extrapolation of the stress values at the integration points. The sum of these values at the exit nodes is denoted as \( \Sigma \text{NFORCE} \), and it is equal to the drawing force, \( F_j \). Then, \( \sigma_y \) is calculated by:

\[
\sigma_y = \frac{F_j}{A_j} \frac{\Sigma \text{NFORCE}}{A_j}
\]

(5)

where \( A_j \) is the cross sectional area at the die exit.

APPLICATIONS AND RESULTS

The adimensional drawing stress value has been calculated by the Upper Bound Theorem and by the Finite Element Method, according to the expressions indicated above (4) and (5), for parameter value range indicated in table 1.

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>( \mu )</th>
<th>( r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>4° - 18°</td>
<td>0 - 0,2</td>
<td>0,20 - 0,40</td>
</tr>
</tbody>
</table>

In the FEM case, several properties of the material used in the process must be introduced in the pre-processing stage. Particularly, the used material is an aluminium which presents a perfect rigid-plastic behaviour and the next characteristics collected in table 2. In the UBT case, this information is not required since the material features are represented in the \( k \) constant.
Table 2: Properties of the material

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density $\rho$</td>
<td>2700 kg/m³</td>
</tr>
<tr>
<td>Young's modulus $E$</td>
<td>$7 \cdot 10^3$ Pa</td>
</tr>
<tr>
<td>Poisson's ratio $\nu$</td>
<td>0.33</td>
</tr>
<tr>
<td>Shear modulus $G$</td>
<td>$2 \cdot 10^3$ Pa</td>
</tr>
<tr>
<td>Yield stress $Y$</td>
<td>$2.8 \cdot 10^3$ Pa</td>
</tr>
<tr>
<td>Hardening parameter $K$</td>
<td>$1.155 \cdot Y$</td>
</tr>
</tbody>
</table>

In figure 3, the obtained results for two different Coulomb friction coefficient values have been shown.

![Graph](image)

Figure 3: Adimensional drawing stress for: a) $\mu=0$; b) $\mu=0.2$

Seeing figure 3 it is possible to affirm that obtained results by the both methods are more close in the without friction case; in particular if the cross-sectional area reduction is low. Exactly, for the studied cases the coincidence intervals are shown in table 3.

Table 3: Coincidence intervals in UBT-FEM cases

<table>
<thead>
<tr>
<th>$r$</th>
<th>$\alpha$ (°)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>$4 - 14$</td>
</tr>
<tr>
<td>0.3</td>
<td>$8 - 18$</td>
</tr>
<tr>
<td>0.4</td>
<td>$12 - 18$</td>
</tr>
<tr>
<td>$\mu = 0$</td>
<td>$6 - 12$</td>
</tr>
<tr>
<td>$\mu = 0.2$</td>
<td>$10 - 16$</td>
</tr>
<tr>
<td>$\mu = 0.4$</td>
<td>$14 - 18$</td>
</tr>
</tbody>
</table>

If friction is not considered, the UBT seems to indicate that, the greater cross sectional area reductions, the greater dies semiangles are necessary, in order to carry out the process under minimum power requirements. However, the FEM seems to indicate the required energy is the same one for all the analyzed semiangle values. When friction is considered, the graphs obtained in both case show that, in general, the necessary energy to carry out the process is smaller if dies with greater angles are used.

CONCLUSIONS

Drawing process has been studied by means of different analytical and numerical methods because of its great importance in the industrial sector. In this work, two method have been used to analyze the process. Concretely, the Upper Bound Theorem and the Finite Element Method.

The first one offers some solutions quite acceptable according to the resources and, besides, it is easy to apply. Its main disadvantage is that it can only be applied under plane strain condition. The second one is a method which gives more exact solutions than the first one. However, it needs more powerful calculation resources and a good knowledge of the FEM code by the users.

The main conclusions extracted from the work are:

- Both methods give similar minimum values of the adimensional drawing stress for each reduction and for each friction condition.
- It is possible to establish die semiangle value intervals in which the adimensional drawing stress values to make a cross sectional area reduction are very next using both methods.
- If friction is not taken into account the UBT and FEM results are more close than it is considered and, in general, they are more low.

As general conclusion it can be said that, if the necessary minimum drawing force to carry out the process wants be calculated, the UBT it is quicker and more economic than the FEM. However, FEM is able to give more information than UBT because it has got better benefits.

REFERENCES


SIMULATION TOOLS
TOOL CONDITION MONITORING IN CUTTING PROCESSES USING HYBRID NEURAL NETWORK BASED SENSOR FUSION STRATEGY

Abderrazzak El Ouafi
Mathematics, Computer and Engineering Department,
University of Quebec at Rimouski, Rimouski, (Quebec), Canada
E-mail: abderrazzak_elouafi@uqar.qc.ca

KEYWORDS
Metal cutting processes, tool condition monitoring, diagnosis, neural networks, sensor fusion.

ABSTRACT
This paper presents a study on tool condition monitoring in metal cutting processes using hybrid neural networks based sensor fusion strategy. Various sensing techniques are combined and used to select suitable monitoring indices. Several models are proposed to establish the relationship between tool conditions and monitoring indices. The proposed approach is built progressively by examining monitoring indices from various aspects and making monitoring decision step by step. Compared to others monitoring schemes, the results indicate a significant improvement and a good reliability in identifying the various tool conditions regardless of the variation in cutting parameters.

1. INTRODUCTION
On-line monitoring is one of the most important components in modern manufacturing. The majority of the processes, when operating near their operational limits, are affected by failures that seriously compromise their reliability and increase the frequency of human interventions. Over the years, researchers have thus agreed on the need of adopting process monitoring strategies to prevent these failures and, when required, detect them as early as possible. In machining process, tool condition monitoring not only reduces the production costs by reducing downtime and unnecessary tool changes, but also improves the product quality. As a result, many approaches have been proposed in the past 20 years (Byrne et al.1995). However, much more research is required to develop a reliable and cost-effective tool condition monitoring system for applications on the shop floor, especially when dealing with variable conditions.

In general, there are three major monitoring tasks, which have been identified in the literature. These are monitoring tool breakage, tool wear and tool vibrations. The majority of the research work conducted so far follows the same framework as that shown in figure 1. It usually consists of three major steps: acquisition of monitoring signals, processing signals and identification of tool conditions. The issues dealing with the understanding of automatic tool condition monitoring systems can be classified according to the development of accurate and reliable online measurement of machining conditions, the selection of the appropriate analysis and processing strategy and the development of signal classification procedures.

The main measurements that have been identified as important indicators for monitoring machining processes (Trusty and Andrews, 1983) include force or torque (Thangaraj and Wright, 1986.), vibration (Hayashi et al., 1988), acoustic emissions (Moriwaki, 1980; Hayashi et al., 1988), and motor current (Altintas, 1992.). Others direct sensing methods have also been reported but their reliability under real conditions is limited. The analysis of data from these measurement systems has demonstrated the difficulties involved in extracting representative characteristics of all process conditions from only one signal. This analysis also shows that two main requirements need to be satisfied when implementing a monitoring system: (i) the measurements must reflect the process behaviour under its normal operating conditions (speeds, feeds, workpiece and tool materials, cutting fluids, etc.) and (ii) the signals must allow discriminating between specific states of the process. Alternatively, integrating several sensors has greatly improved the quality and reliability of condition identification (Colgan et al., 1994). This type of approach remains nevertheless tributary to the methodology used to develop the processing and the classification methods.

The analysis and processing strategy is related to monitoring indices extraction. The processing is generally preceded by signal conditioning. Typical monitoring indices include: time-domain indices, time-frequency domain indices, higher order spectrum indices and wavelet indices. For tool condition monitoring, one of the best-known approaches consists in monitoring amplitude increases or variations in the signal (Inasaki, 1985). However, such techniques are problematic since variations are strongly contaminated by noise and quickly become difficult to interpret. The introduction of new processing techniques has improved the efficiency of the detection, particularly for tool wear and tool breakage (Takata, et al. 1985; Barker et al., 1993). The spectral analysis method is one of the most commonly used signal processing technique in tool breakage detection. This technique has a good resolution only in the frequency domain but a very bad resolution in the time domain. Also some signal information in time domain is lost in the spectral analysis process. The precision and speed needed for metal cutting applications require more efficient approaches both in terms of instrumentation and analysis.

The classification procedure consists of using monitoring indices to classify the current tool condition based on predefined conditions. When dealing with a specific failure mode, the detection procedure consists of using signature features, which is sensitive to this particular failure mode.
The classification methods can be divided into two categories: weighting methods, including pattern recognition, fuzzy logic, and artificial neural networks; and decomposition methods, such as decision trees and knowledge-based systems. For specific failure mode detection, the majority of research efforts are concentrated on acoustic emission and cutting forces. These techniques are sensitive to tool failures but require extensive calibration. Recently, attempts have been made to use sensor fusion to generate the required signature features using information from multiple sensors (Rangwala and Dormfeld, 1990). Other efforts have been made in the development of signal classification procedures for automatic identification of several defect classes. Artificial intelligence techniques are rather attractive since they offer the ability to deal with the substantial degree of uncertainty, which is characteristic of the cutting process (Guillot and El Ouaffi, 1992). Results obtained using pattern recognition methods indicate an interesting reliability for detecting tool failure under fixed conditions. For automatic tool condition monitoring in metal cutting applications, however, these signal classification methods must be reliable for changeable cutting parameters and variable conditions.

Fig. 1. Diagram of the setup for tool condition monitoring

This paper proposes a new automatic monitoring approach for on-line tool condition identification in cutting processes. The considered tool conditions include normal cutting, air cutting, transient cutting, tool breakage, moderate and severe wear, chatter, changes of cutting parameters, sensors malfunction and process noise. The proposed approach is based on hybrid neural networks based sensor fusion strategy. Various sensing techniques are combined and used to select suitable monitoring indices. Forces, vibration, acoustic emissions and motor current sensors are used in this study to generate a set of signature features, which characterize the various tool conditions. A total of 16 monitoring indices are selected to describe the signature characteristics of 10 various tool conditions. Several models are developed using a powerful and systematic model optimization procedure in order to describe the relationship between the tool conditions and the monitoring indices. A classification scheme is then developed and tested under various cutting conditions.

2. THE PROPOSED MONITORING APPROACH

The suggested approach follows basically the steps described previously. The signals are acquired from the various sensors, pre-processed and then processed in the time domain by several ANN models to identify process conditions. As illustrated in Figure 2, during each control step, signals obtained from the «p» sensors are used to extract «q» signature features representing «r» monitoring indices created by the most recent successive «s» samples. Then, the «r» monitoring indices are forwarded into the ANN model that generates the classification code identifying the «t» various tool conditions.

Figure 2 - The identification approach - Stationary time-domain pattern classification

Machining is a complicated dynamic process with various nonlinearities and stochastic disturbances. In these conditions, because the deterministic models are typically valid only for limited range of cutting conditions, the ANN models present the best modelling alternative. Although various neural techniques can be used in this approach, the radial basis function network (RBFN) model seems to be one of the most appropriate because of its flexibility. RBFN networks are non-linear hybrid neural network in that it has the same structure of layers as the MLPs neural network, but a quite different information process (the data is clustered) in the hidden layers, which increases the speed of solution.

The difficulty to implement such an approach lies in the choice of sensors and the modelling conditions. These choices represent the two basic ingredients of any sensor fusion technique. Selecting the model form and modelling technique is not sufficient to produce the best model. Although hybrid neural network modelling can improve the identification process, several parameters that greatly influence model quality remain to be defined. In particular, these parameters include sensor selection, choice of the monitoring indices and modelling conditions.

The literature has no mention about established method to systematically develop this type of approach. However, it is reasonable to assume that sensor integration is carried out through a series of steps in which decisions are made based on specific statistical test results. These tests depend on the correlation between modelling parameters and characteristics to be identified. In order to extract the best combination of parameters, an experimental design method is used. Based on the orthogonal matrices, this technique significantly reduces the number of combinations to be tested when there are many parameters and it is impossible to evaluate all potential combinations.
Variable model selection is based on the analysis of the effect of each parameter on the model quality as well as the parameter contribution to decrease modelling and identification errors.

The proposed method can be summarised by the following steps: (i) collect data to train and verify the models (all parameters and conditions that may influence process must be identified and considered. For validation, only a limited number of tests are required); (ii) select the modelling technique and performance criteria; (iii) select a suitable experimental matrix to design the required models (rows of the matrix corresponding to models and columns representing which variables are to be included in each model); (iv) train the generated models and evaluate their performances according to the established criteria; (v) determine the effects of each variable and evaluate its contribution in each performance index, and finally, (vi) determine the optimal model configuration.

3. MONITORING APPROACH SIMULATION

Data collection

Before testing the proposed tool condition monitoring approach by simulation, models training are mandatory. For this task, signals of the cutting process under various conditions must be produced and typical signature features extracted. In this study, experimental tests are performed on a 34 KVA Mori Seiki SL-25M turning center to manufacture titanium alloy and aluminium alloy parts. The carbide cutter inserts are mounted on a bracket on which vibration and acoustic emission sensors are attached. The set-up is mounted on a Kestler dynamometric table that is located on the turret. During the experiments, forces, vibrations, acoustic emissions, electric spindle current and electric feed current signals are simultaneously monitored.

A successful tool condition monitoring must be sensitive to tool change in tool condition and insensitive to cutting conditions variations. Hence, a total of 342 cutting tests conducted under 54 various cutting conditions are carried out in two phases. The first one covers normal machining operations using variable cutting conditions. The second phase covers operations during which undesirable tool conditions are caused deliberately. Thereafter, the obtained signals are used to build the signature patterns representing normal operating conditions as well as failure conditions.

Based on the sensor signals, monitoring indices are used to describe the characteristic features of different tool conditions. It has been stated that the monitoring indices selection is an art, while the literature has thoroughly developed the classification techniques. In this study, emphasis has been placed on the rational choice of monitoring indices rather than using other methods such as curve fitting, probability of error and correlation coefficients. A systematic analysis of sensor signals monitored under various tool conditions as well as a review of pertinent literature provided the basis of the choice of monitoring indices. The selected indices are: cutting forces (\(F_x, F_y, F_z\)); vibrations (\(V_x, V_z\)); acoustic emission; mean of resultant cutting force (\(mF\)); crest factor of cutting forces (\(cF_x, cF_y, cF_z\)); ratio of cutting forces (\(rF_x, rF_y, rF_z\)); spindle motor current (\(SMC\)) and feed motor currents (\(FMC_x, FMC_z\)). These indices represent the most important characteristic features of the monitoring signals in time domain.

Modelling investigation

The methodology used to design the model can be summarised by the three following points: (i) design and train a sufficient number of models, each one is developed in order to preserve the orthogonality of the experimental matrix; (ii) calculate the performance indices for each model; (iii) determine the effects of each parameter on model quality.

The parameters, which may potentially influence the quality of the models, are grouped and used to design a modelling experiments matrix. The selected matrix suggests that 32 models are required to allow the analysis of the effects of the modelling parameters on the models quality as well as the contribution of each parameter to the variation of the characteristics to be evaluated. These characteristics are: learning error (\(I_{LE}\)), identification error (\(I_{IE}\)), total error (\(I_{TE}\)), normal cutting identification error (\(N_{CE}\)), air cutting error (\(A_{CE}\)), transient cutting error (\(T_{CE}\)), tool breakage error (\(T_{BE}\)), moderate wear error (\(M_{WE}\)), severe wear error (\(S_{WE}\)), chatter error (\(C_{CE}\)), changes of cutting parameters error (\(C_{PE}\)), sensor malfunction error (\(SM_{SE}\)) and process noise error (\(P_{NE}\)).

Modelling analysis

In order to test the validity of the proposed monitoring approach, three procedures are designed. In the first procedure, all the samples are used as learning simples and also validation samples. In procedure 2, half of the samples are randomly picked as learning samples, while the remaining samples are used as validation samples. In procedure 3, the samples obtained under certain cutting conditions are used as learning samples while the remaining samples obtained under different cutting conditions are used as validation samples. In the majority of previous studies, modelling performances are evaluated using only the first procedure.

The three procedures are combined to two other methods selected to evaluate the effect of each modelling parameter on the quality of the models. The first one involves evaluating the parameters effects on the learning error (\(I_{LE}\)), the identification error (\(I_{IE}\)) and the total error (\(I_{TE}\)). The second one evaluates specifically each parameter effects on the quality of the identification of each tool condition. In both cases, the evaluation is based on model statistical characteristic analysis to determine its precision and also to highlight each parameter effects and contribution to residual error variation.

According to the first method, the results demonstrate that the proposed method works well in identifying normal cutting, air cutting, transient cutting, tool breakage, chatter, changes of cutting parameters, sensors malfunction and process noise tests, but performs rather poorly for moderate and severe wear. A careful examination indicates that this is because a number of
moderate tool wear samples were mis-classified to normal cutting and severe tool wear samples were mis-classified to tool breakage.

Analysing the effect of the variables on the models quality and their contributions in reducing the \( l_{\alpha} \), \( l_{\beta} \), and \( t_{\alpha} \) errors make possible the identification of three quasi-optimal configurations. The first version is achieved by setting each parameter at a level, which minimises the learning error. The second version is obtained by choosing the parameters, which minimise the identification error. Setting the parameters in order to minimise the total error provides the third model. The comparisons of the three optimal models versions demonstrate very similar performances in terms of learning and identification errors.

Following this evaluation, the option of using only one sensor at the time as well as using cutting speed, depth of cut and feed rate, as additional monitoring indices were examined. Accordingly, the results reveal that the monitoring indices based on cutting forces and power signals are more reliable than those based on vibration and the acoustic emission signals in detecting tool conditions. However, combining all monitoring indices improves the model quality. Furthermore, it is also confirmed that the classification can be improved when using cutting conditions as additional monitoring indices. The results obtained with this evaluation scheme demonstrate that the proposed approach works well in identifying normal cutting, air cutting, transient cutting, tool breakage, severe tool wear, chatter, changes of cutting parameters, sensors malfunction and process noise, However, it performs rather poorly for identifying moderate tool wear which is mis-classified to normal cutting.

The model obtained when following this second evaluation scheme cannot be improved using the same monitoring indices. The errors arise from the model inability to constantly and efficiently detect all tool conditions. This is where the third evaluation scheme becomes interesting. Indeed, using specialized models that have fewer tool conditions to identify, it is possible to establish several models operating in parallel, and each one specialized in detecting one failure group. In order to identify the favourable conditions required to establish specialized models, an in-depth analysis evaluates the ability of all models to detect each type tool condition by classifying them according to \( l_{\alpha} \), \( l_{\beta} \) and \( t_{\alpha} \) criteria.

These results highlight some rather revealing observations. The tendency to classify tool conditions by group is easily apparent. Indeed, if one considers the \( l_{\alpha} \) as the principal classification criterion, the same models generally react favourably to each failure mode. This concept of tool condition family is also confirmed by the observation of \( l_{\beta} \). In most cases, it is also the same models that react unfavourably to each tool conditions group. Based on these observations, tool conditions are grouped in three groups to produce three specialized models: \( N_{m} \) to identify normal cutting, air cutting, transient cutting and changes in cutting parameters; \( W_{m} \) to detect, chatter, process noise moderate tool wear and severe tool wear; \( F_{m} \) to detect tool breakage and sensors malfunction. It is important to note that each model is designed and trained to detect only one tool condition group. The results are promising but are not as accurate as the model produced with the second evaluation scheme. However, using the three models result in a greater total error than the error obtained from one global model.

4. CONCLUSION

This paper has presented a systematic hybrid neural network based sensor fusion strategy to support a monitoring approach for an on-line cutting processes condition identification. Several models are developed using a powerful model optimization procedure in order to describe the relationship between tool conditions and monitoring indices. A classification scheme is then developed and tested under various cutting conditions. Built progressively by examining monitoring indices from various aspects and making decision step by step, the proposed approach offers the ability to express the effect of each modelling parameter on the performance and the efficiency of the monitoring system. Using an example applied to turning, the paper demonstrates the possibility of establishing a general model able to identify not only specific failure mode but also detect and identify various tool conditions and defect classes under various process conditions. Compared to several available monitoring schemes, the performance of the proposed approach indicates a significant improvement and a good reliability regardless of the variation in cutting parameters.

5. REFERENCES

Byrne, et al. 1995. "Tool condition monitoring (TCM) - the status of research and industrial applications", Annals of the CIRP, 44.
SELECTED PROBLEMS OF EFFECT ANALYSIS AND SAFETY ASSESSMENT OF COMMANDS GIVEN BY THE OPERATOR OF THE TECHNOLOGICAL DEVICE USING ARTIFICIAL NEURAL NETWORKS

Wojciech Kacalak and Maciej Majewski
Department of Mechanical Engineering
Technical University of Koszalin
Racławicka 15-17
75-620 Koszalin, Poland
E-mail: {kacalakw, mmaj}@tu.koszalin.pl

KEYWORDS
Speech, voice communication, artificial intelligence, neural networks, computer engineering applications, commands.

ABSTRACT
The voice communication between technological devices and the operator becomes a stronger challenge as technology becomes more advanced and complex. New applications of artificial neural networks are capable of recognition and verification of effects and safety of commands given by the operator of the technological device. In this paper there is a review of the selected issues on estimation of results and safety of the operator’s commands as well as supervision of the technological process. A view is offered of the complexity of effect analysis and safety assessment of commands given by the operator using neural networks. There is also an intelligent two-way voice communication system between the technological device and the operator presented, which consists of the intelligent mechanisms of operator identification, word and command recognition, command syntax and result analysis, command safety assessment, technological process supervision as well as operator reaction assessment. The paper presents some research results of the simulation of the command effect analysis and command safety assessment subsystems. The first part of the paper introduces a new concept of modern supervising systems of the technological process using an intelligent layer of two-way voice communication between the technical device and the operator and discusses the general topics and issues. The second part is devoted to a discussion of more specific topics of the automatic command verification that have led to interesting new approaches and techniques.

INTRODUCTION
In the future, voice messages in natural language will undoubtedly be the most important way of communication between humans and machines. Great progress is made in many fields of science, where communication between the technological devices and the operator is an important task, e.g. motorization, road traffic, etc. Applying an intelligent layer of a two-way voice communication system between the technological device and the operator is becoming a more important task. The goal is to reach a higher level of the intelligent process supervision systems provided with the intelligent two-way voice communication system between the technological device and the operator to increase efficiency and humanization of the production process (Kacalak and Majewski 2003b).

The proposed intelligent layer of two-way voice communication between the technological device and the operator is presented in Figure 1. It is equipped with the following intelligent mechanisms: operator identification, recognition of words and complex commands, command syntax analysis, command result analysis, command safety assessment, technological process supervision, and also operator reaction assessment (Kacalak and Majewski 2004).

If the operator is identified and accepted by the system, a produced command in continuous speech is recognized by the speech recognition module and processed to the text format. Then the recognised text is analysed with the syntax analysis subsystem. The processed command is sent to the word and command recognition modules using artificial neural networks to recognise the command, which next is sent to the effect analysis subsystem for analysing the status corresponding to the hypothetical command execution, consecutively assessing the command correctness, estimating the process state and the technical safety, and also possibly signalling the possible error caused by the operator. Then the command is sent to the safety assessment subsystem, for assessing the grade of affiliation of the command to the correct command category and making corrections. Next the command execution subsystem signalizes commands accepted for executing, assessing reactions of the operator, defining new parameters of the process and run directives. The subsystem for voice communication produces voice commands to the operator.
Figure 1: Scheme of the Intelligent Layer of Two-Way Voice Communication between the Technological Device and the Operator

The task of the intelligent system of two-way voice communication between the technological device and the operator, shown in Figure 2, is generating voice messages to the operator about the process state, storing the operator's voice commands, analyzing reaction time of the operator, analyzing syntax and assessing safety of the operator's commands, and also estimating the operator's reaction. That system in co-operation with the comprehensive supervision system will also make it possible to analyse the state of the process corresponding to the hypothetical execution of the operator's command.

Figure 2: The Intelligent System of Two-Way Voice Communication between the Technological Device and the Operator
A NEW CONCEPT OF MODERN SUPERVISING SYSTEMS OF THE TECHNOLOGICAL PROCESS

According to the new concept of the modern supervising systems of the technological process, the operator may stay away from the technological device in another room, or even in any distance from it (Kacalak and Majewski 2003a). There is no need of an operator being present at the work stand by the technological device. It is made possible by using the intelligent system of two-way voice communication between the technological device and the operator (O'Shaughnessy 2000; Jurafsky and Martin 2000).

In the future in the production systems the operator will be separated from the technological devices with the intelligent layer, which will provide the system with more resistance from the operator's errors and much more efficient supervising of the process (Weinschenk and Barker 2000). The task of the supervising system of the technological process, shown in Figure 3, is to provide with automation and a high level of effectiveness of the process supervision with an elastic and controlled co-operation of the operator observing directly or in a distance. The comprehensive supervising system of the technological process is equipped with the self-learning and self-control mechanisms.

Creation of the modern supervising systems of the technological process using the assumption mentioned above will make it possible to eliminate scarcities of the typical co-operation between the operator and the technological device (Kacalak and Majewski 2003a).

---

**EFFECT ANALYSIS AND SAFETY ASSESSMENT**

The recognised command given by the operator is processed and sent from the command syntax subsystem to the verification subsystems of effects and safety. The effect analysis module, shown in Figure 4, makes analysis of the recognised command. The technical safety of the technological device is checked by analysing the state of execution of the commands required to have been done as well as the commands to execute in next decisions.

---

**Figure 3: General Scheme of the Supervising System of the Technological Process Using the Intelligent Layer of Two-Way Voice Communication System between the Technological Device and the Operator**

---

**Figure 4: The effect analysis module**
The process parameters to be modified by executing the command are checked and the allowable changes of the parameter values are determined. The analysis of the parameter values is based on the technological process features. The values of the parameter changes are the input signals of the neural network of the process state assessment system. The neurons of the neural network represent solutions to the diagnostics problem. The neural network also makes an estimation of the grade of safety of the recognised command.

![Figure 5: Technical safety system](image)

The technological safety assessment system, shown in Figure 5, is based on a neural network which is trained with the model of work of the technological device. New values of the process parameters are the input signals of the neural network (Principe et al. 2000). As the work result of the system, voice messages from the technological device to the operator about the possibility of executing of the command are produced.

![Figure 6: System for checking the state of the automatic technological device](image)

RESEARCH AND SIMULATION RESULTS

Research results of the simulation of the command effect analysis and command safety assessment subsystems make it possible to draw several conclusions. The system for checking the state of the automatic device for grinding of small ceramic elements before the next command is executed, shown in Figure 6, is a very important element of the technical safety system.

![Figure 7: Neural network simulations of the technological process model](image)

The simulation set of the technological device diagnostics and the process state assessment, built for creating and training artificial neural networks is shown in Figure 7. The neural networks are trained with the model of the technological process.
The ability of the neural network to learn to recognise specific process states depends on the number of learning epochs. The specified time of learning enables the network to minimize the error so that it could work more efficiently. Based on the research, the following conclusion has been reached as shown in Figure 8. The error is about 20% at learning time equals 50 epochs and 5% at 100 epochs. The error has been dropped for about 90% after training with 60 series of all patterns.

![Figure 8: Neural network error rate](image)

CONCLUSIONS AND PERSPECTIVES

The condition of the effectiveness of the presented intelligent two-way voice communication system between the technological device and the operator is to equip it with mechanisms of command verification and correctness. In the automated processes of production, the condition for safe communication between the operator and the technological device is analyzing the state of the technological device and the process before the command is given and using artificial intelligence for assessment of the technological effects and safety of the command. In operations of the automated technological processes, many process states and various commands from the operator to the technological device can be distinguished. A large number of combined technological systems characterize the realization of that process. In complex technological processes, if many parameters are controlled, the operator is not able to analyse a sufficient number of signals and react by manual operations on control buttons. The research aiming at developing an intelligent layer of two-way voice communication is very difficult, but the prognosis of the technology development and its first use shows a great significance in efficiency of supervision and production humanisation.

REFERENCES


BIOGRAPHIES

WOJCIECH KACALAK was born in Zdunskia Wola, Poland. He studied Mechanical Engineering at the Technical University of Lodz, where in 1970 he obtained his master’s degree, in 1974 doctor’s degree and as of 1989 he has been professor of technical sciences. Since 1970 he has been working at the Mechanical Faculty of the Technical University of Koszalin doing research on optimization of manufacturing processes with the application of neural networks and fuzzy logic.

MACIEJ MAJEWSKI was born in Polczyn-Zdroj, Poland. He studied Applications of Computers in Engineering at the Technical University of Koszalin (MSc BSc degree in Application and System Design in 2000). He also studied at the University of Granada (Spain) (1 year Erasmus grant) and was given professional training by the European Commission DG in Luxembourg in 2001. He is a Ph.D. student at the Faculty of Mechanical Engineering of the TUK.
AN INFORMATION PROCESSING TOOL FOR THE MANAGEMENT OF NATURAL WATER SYSTEMS

Naghmeh Keshavarzi-Roonizi
Rahim Ghorashi
Vahid Nassedi
Department of Chemical Engineering
Loughborough University
Loughborough, LE11 3TU, UK
Email: n.keshavarzi-roonizi@lboro.ac.uk

KEYWORDS
Information Processing Tool, Hydroinformatics, Network System.

ABSTRACT
Fast and efficient access to up-to-date information is the most important factor in enabling organisations to make appropriate decisions in dealing with day-to-day work or long term policy matters. In particular, the effective monitoring and management of vast and complicated natural systems such as rivers is not possible without processing large amounts of data in short periods of time. Therefore, the use of computer models which reduce the cost of acquisition and increase the speed of data handling has become an indispensable feature in the management of river/estuary systems. However, due to the complexity of the models used to simulate the hydrodynamics, pollutants dispersion and interaction between water flow and man made structures, the utilisation of computer models of river networks requires expertise and significant amount of experience. In this paper the development of an IT system which enables operators with minimal experience in hydrodynamic modelling to access and process information related to the behaviour of complex natural water systems is presented. This system provides a valuable tool for the management of river/estuary networks specifically in dealing with critical events such as flooding. Implementation of the IT system described here is demonstrated via presentation of sample applications.

INTRODUCTION
Natural water systems are complex flow domains where water quality is affected by the interactions of a wide range of hydrodynamic, geometrical, physico-chemical and biological phenomena. Consequently, computer models developed to simulate hydro-environmental conditions in these systems have inherently elaborate architectures and usually lack flexibility. In particular, in systems such as rivers, estuaries and coastal zones, where crude, large scale averaging is not acceptable, the modelling is based on sophisticated computer programs whose utilization require expert level training. This often precludes the use of these models by operators in organisations such as environment agencies, local authorities and water companies.

At present, therefore, the use of the most accurate and reliable hydro-environmental simulation tools are restricted to research centres. To overcome this difficulty the relatively new discipline of hydro-informatics has been developed in recent years (Klosgen and Zytkow, 2003). The idea is to utilise maximum capabilities of modern computing systems in conjunction with innovative programming algorithms, combined with the state-of-the-art mathematical and numerical schemes to enhance the level of software intelligence so that operators with minimal training can generate useful simulations in a very short time.

Network strategies play an important role in the applicability of hydroinformatic interfaces. They enable the users to interact, communicate and run specific application modules simultaneously from different locations. They also allow resource sharing, parallel computation, checking and implementation of operation redundancy protocols for users at geographically distant locations from each other.

The main purpose of the present work is to develop an Information Processing Tool (IPT) for data processing, information handling and, in general, management support, based on the strategies developed for hydroinformatic applications. Industrial relevance of the proposed strategy and economic implications of its availability are demonstrated by presenting a case study.

IPT SYSTEM ARCHITECTURE
To achieve the goals of ease of use, clarity, practicability, speed, multi-user capability and computing economy the described Information Processing Tool is constructed as a combined system consisting of three distinct modules. The overall architecture of the developed system is shown schematically in figure 1.

The modules shown in figure 1 can be briefly defined as:

• Front-end, provides a network node for communication between the users and software assets that are in the processor (data generator) and back-end modules.
Figure 1. Schematic of the IPT System

- Data-generator, is a systematic library of various modelling software that use deterministic or stochastic techniques to simulate, evaluate and generate quantitative results for hydro-environmental problems for which the user seek solution.

- Back-end, provides a network node for the import and processing of the data (i.e. results, answers, evaluations) generated by the software in the generator module. The processed results are logged and returned to a window in the front-end for the utilization by the users.

In the following section the basic structure and operations of these modules are described.

**Front-End Module**

The front-end of the system is structured as an interactive interface using the following principles and techniques:

- A user adaptively techniques to allow multi-user capability.

- User logging that allow the system to maintain a record of users using the system at any given time or the users who have used the system within a set period of time.

- Categorisation capability to record users, organise required applications and produced responses into clearly identified groups for efficient manipulation of the computer resources. Predetermined relations between various input factors are incorporated into the interactive interface for purposeful grouping of related items and disentanglement of unrelated questions etc. This is a crucial aspect of our user interface because the aim has been to assist the general users, who may not have expert knowledge concerning sophisticated modelling software incorporated in the data-generator module, to obtain meaningful analysis results even if the questions that they pose may not be entirely clear (from a mathematical or scientific point of view).

- A feature of the interface is to maintain ‘Simplicity’. Communications are kept simple and clear using user's own language to provide shortcuts meaningfully related to more complex procedures.

- All of the available options and procedures to perform a given task are made visible without distracting the user with extraneous or redundant information.

- The user is supplied with feedback at each stage of communication with the front-end interface. This is to guide the user to give consistent input and choose appropriate options necessary for the efficient implementation of a required task.

- The front-end interface has a high degree of tolerance to reduce cost of mistakes by allowing undoing, redoing, cancelling and going back-while preventing errors wherever possible by tolerating varied inputs and sequences and by interpreting all logical actions reasonably.

**Data-Generator Module**

This module is the main engine of the present information processing tool. Essentially it consists of a library of various mathematical models for the simulation of hydrodynamic and pollutant mixing and transport problems in networks of tidal and non-tidal rivers. The module is flexible and can incorporate both in-house developed models or public domain and commercial software. The design of this module is such that after the evaluation of all aspects of a presented problem, (i.e. available input, questions needed to be answered, desired quantitative results identified) the most suitable model for its solution from the available models in the data-generator module is selected. The interface feeds the input data to the selected model initiating numerical computations. At the end of the computations the simulation results are passed to the back-end module for further processing required before presentation of the answers. Due to the extensive and varied nature of problems encountered in natural water systems. It is impossible to develop a model universally applicable under all conditions. Therefore in practice a plethora of different models and strategies have been constructed. The data-generator embedded in the present system is capable of dealing with following problems:

- Hydrodynamic modelling of long and narrow non-tidal rivers and branching river networks. The required information in these problems is to quantitatively predict flow velocity, pattern of currents and water depth in relation to rain fall, surface run-off, water seepage through bed and river banks. These are the necessary data required in planning agricultural, irrigation, water
extraction projects and construction of flood defences.

- Prediction of the distribution and concentration of chemically reactive and inert solute pollutants introduced into long and narrow non-tidal rivers and branching river networks.

- Hydrodynamic modelling of wide tidal rivers (estuaries) including the prediction of extent of flow channel during the flood and ebb tides along the water course. This information is required in the planning of optimum effluent discharge policies in such hydro-environmental systems. Moving boundary of the flow channel and its continuous change during a tidal cycle is a particularly difficult problem and depending on topological, physical and hydrodynamic features of a given water system a different simulation technique should be adopted. For this reason, in the present module five different models for performing this task is incorporated. The system selects the most suitable option via communications with the user and evaluation of the particular conditions of a given problem. Details of these models have been published before and will not be repeated here (Nassehi and Williams, 1986; Nassehi and Bikangaga, 1993; Nassehi and Kafai, 1999; Passone et al, 2001). Evaluation mechanism is based on a Case Based Reasoning Algorithm (Passone, 2002).

- Prediction of the distribution and concentration of chemically reactive and inert pollutants introduced into estuaries.

- Specific features such as prediction of conditions resulting in ice sheets covering a non-tidal river in severe winters.

- Prediction of conditions in a planned fish farm and similar commercial exploitation of river systems.

As mentioned before some of the models incorporated in the present data-generator are in-house developed (details are published previously) others are public domain software. The most notable example of an imported software used in the present system is a package called RIVER2D (Steffler and Blackburn, 2002). This package is particularly suited for the analysis of the effects of ice cover in river hydrodynamics and can also be used to quantitatively analyse fish farm specifications. It uses a finite element scheme based on the Stream-line Upwind Petrov-Galerkin method (Nassehi, 2002) to generate simulations for both sub-critical and supercritical flows and includes a facility to obtain solutions for wet-dry area problems.

Interactions between the front-end and data-generator modules is a necessary feature of the present Information Processing Tool because only using a clear and systematic procedure the user can be guided to obtain the best possible simulation for complex problems without lengthy training and experience in computer modelling.

**Back-End Module**

The back-end of the system provides secondary services (post-processing) necessary in the analysis of complex problems. It should be noted that the simulations results obtained via the data-generator mainly consist of large amounts of numerical data. These tables of data are usually very difficult to understand or interpret in a short time in terms of physical descriptions required for making engineering or management decisions. Secondary processing of data or post-processing converts the simulation results to easily understandable forms. It also eliminates noise and parasitic output to prevent confusion. In a large number of situations instead of tables of numerical data the results are shown as graphs or contour plots of specified variables. The back-end module can also communicate directly with the front-end at the preliminary stages of an application (i.e. problem simulation). For example, it is designed to provide answers regarding the validity of a given input on the basis of its self-consistency or logical status. This feature is mainly utilized to guide the untrained user to obtain required information from the system with ease and efficiency.

**IMPLEMENTATION OF THE DEVELOPED IPT ON A NETWORK**

In most cases management of natural water systems consists of a variety of performed by a team of operators rather then individuals. Members of such a team my be based at stations geographically far from one another. The overall decision may also depend on considerations related to different aspects of a problem such as preservation of environment, economic viability of a plan and other industrial factors. Therefore the described system can only be regarded as an effective practical tool if it can be operated on a network with easily accessible multi-user capability. It should also be possible to readily extend the system to include new parts and capabilities in each of its modules. To satisfy these criteria the constructed IPT is based on a network architecture which provides:

- Functionality- it allows users to meet their job requirements. The network also provides user-to-user and user-to-application connectivity with reasonable speed and reliability.

- Scalability- the network is able to grow. That is, the initial design should grow without any major changes to the overall design.

- Adaptability- the network is designed in anticipation of future technological developments and hence it does not include any element that can limit implementation of new technologies as they become available.
To achieve the stated goals currently we use WEB based access to provide networking for our multi-user system.

**SAMPLE APPLICATION OF THE SYSTEM**

In order to demonstrate the operation of the described IPT we consider the problem of flood routing in a long and narrow non-tidal river reach. The practical example of such a problem is the prediction of the extend of flooding in upper parts of river Tees in Northumberland (north east of England, UK). Economic and industrial implications of ability to accurately simulate flooding scenarios in a well developed region such as upper Tees basin is self-evident. From a hydro-engineering point of view such a prediction is not a trivial matter and computer simulation is considered to be the best possible technique to obtain the necessary quantitative information.

Using the IPT system after the identification of the problem the communication with the software starts. In what follows the procedure of using the system via the front-end, information provided by the system, recommended data-generator model, implementation of the model and the obtained simulation are shown in figure 2.

**Task Specifications & Purpose:**
- The task is required:
  - To make management decisions.
  - or
  - To obtain data for future developments of the system (e.g. to construct flood barriers along the reach).
  - or
  - To generate data for the expansion of the system?
  - Is this the first time that you have asked for this task to be performed?
  - Should there be user-user communication concerning this task?
  - etc.

**Input Data Evaluation:**
The data-generator checks the given information for consistency and logical accuracy. If needed further questions are asked. It then searches the data-generator to identify the most suitable modelling software to perform the task. In the presented example this is RIVER2D.

**Select the Required Task:**
- Non-tidal river hydrodynamics
- Flood routing
- Tidal river hydrodynamics
- Solute dispersion in estuaries
- etc.

chosen: Flood Routing

**Connect the user to the selected model (application):**
The user is guided through specific steps required for running RIVER2D. Appropriate facilities (e.g. windows) for the insertion of input data in a format acceptable to the package are made available. In the case of RIVER2D the basic steps are shown in figures 3 and 4.

**RUNNING OF RIVER2D**

After the selection of the RIVER2D as the most appropriate modelling tool for the required task the system open a window as shown in figure 3. The user is asked to input the basic outline geometry of the river reach where flood routing is required. This prepared file is labelled 'name.bed'. An appropriate coordinate system has been given in the opened window and the points defining the geometry can be selected using the mouse.
After the definition of the .bedfile the computational grid required for numerical calculations is simply via giving a specific command (see figure 4). The generated grid is displayed to the user as an interactive file making it possible to be modified by altering ‘grid density’, ‘number of elements’, and ‘addition’ or removal of various computational nodes.

After the generation of computational grid the package proceeds to the simulation stage. The results, in this case water depth at different locations in the selected reach are shown as a contour plot (figure 5). The presented contours clearly show the accumulation of water at the lower parts of the reach and the user can immediately identify areas locations where water levels may rise above the banks causing flooding.

CONCLUSIONS

A versatile information processing system for dealing with complex hydro-environmental problems have been described. General architecture of the system and the criteria used in its design are explained. The main advantage of such a system is that it can be used by operators with minimal training and specific expertise in hydro-environmental modelling. An example is given in the paper which shows how a relatively complicated hydrodynamic problem such as flood routing under realistic conditions for a real river situation can be readily carried out by the application of the present system.

ACKNOWLEDGMENT

The authors wish to thank Engineering and Physical Research Council of UK (EPSRC) and Loughborough University for PhD studentships which enabled the development of the Information System described in this work.

REFERENCES

http://bertram.civil.ualberta.ca/software/River2D.pdf
FUZZY CONTROL IN AGRICULTURE:
SIMULATION SOFTWARE

Calixto Escobar
Industrial Technical Engineer in Electronics by the
University of Malaga, Spain. calixto@auna.com

José Galindo
Department of Computing Languages and Sciences,
University of Malaga, Spain. ppgg@lecc.uma.es

KEYWORDS
Simulation software, Simulation in agriculture, Fuzzy
controller, Controller based on fuzzy knowledge, Fuzzy
controller software, Rule based system, Agriculture.

ABSTRACT

In the present document the software called “System of
Fuzzy Control (SCD)” is described as the platform that
allows to simulate some types of control systems based
in fuzzy logic. This software allows the user to create
and modify the elements that make a fuzzy knowledge-
based controller in an intuitive and clear way. The
knowledge base of SCD is formed by a set of rules of the
conditional type <IF... THEN...> being possible with the
choice of connective operators and negation between
antecedents, which are fuzzy comparisons. The program
is very flexible, visualizing in a graphical form the char-
acteristics of each fuzzy variable next to the simulation
data. This software can be applied to any scope of the
industrial simulation. In our case we have applied it to
the climatic control on industrial greenhouses.

1. INTRODUCTION

Ever since Mamdani [9] showed the first application of
the fuzzy logic to the control of a particular process, an
extensive range of applications, from household-electric
to robots and systems of industrial control [6,7,11] have
been developed under the premises of the fuzzy control.
At the present time we can find a great variety of soft-
ware [1,12], which is able to control processes being
based on its structural approach to the form of the human
thought and to the natural language of an expert oper-
tor.

The software that we present here tries to be a contrib-
ution with substantial improvements with respect to
other contributions [3]. It covers the purpose of being
able to create, in the most possible general form, fuzzy
knowledge-based controllers. In addition, the application
to the fuzzy logic to the agricultural production systems
[2, 4, 10] is an appropriate approach in the search of
solutions because it would serve like a tool for “storing”
expert knowledge of aid to the decision. For this reason
we have applied the program to the crop of peppers in a
greenhouse in Almeria (Spanish Southeastern).

In the first place we will expose a description of
software by listing its main characteristics. Next we will
show the phases of creation of the system of climatic
control in the application. Once the system is created
we will come to simulate, in order to do so, we will see
the different phases that the application makes to obtain
the control actions. Finally, we present some conclusions,
future works and references.

2. SCD SOFTWARE

SCD [5] is a generic platform for the development and
simulation of systems of fuzzy control, based on fuzzy
knowledge. The program allows to evaluate the system
developed for a sequence of input values and to obtain
the corresponding sequence of control action (output).

The menu Variable provides the options on the input
and output variables (creation, modification, elimination
or undo). Equally, the menu Rules allows to add, modify,
eliminate, recover or organize the rules in the knowledge
base. Simulation options are in the menu Inference. We
will comment them later in Section 4.

The main characteristics of SCD are:
- Redimensionable in run time in the number of inputs
  variables, outputs, rules, fuzzy sets of each
  variable and antecedents, consequents in each rule.
- Fast definition of variables, rules, labels...
- 7 types of Fuzzy Sets: Gamma, L, Singleton, Interval,
  Triangle, Trapezoidal and Extended Trape-
  zoidal.
- Universe of Speech: Option of maintenance of the
  proportionality of the fuzzy sets previously defined
  (or to maintain them fixed) after modifications of the
  Universe.
- Extended Trapezoidal: This novel type of fuzzy
  set is a linear function in some pieces, that allows
  great flexibility (even for non convex sets). It is
  redimensionable in the number of points.
- 5 Types of Fuzzifications: Singleton, Interval, Tri-
  angle, Trapezoidal and Extended Trapezoidal
- Precise Analysis and per Sequence of States.
- 12 Types of Defuzzifications: Average of Maxi-
  mums (AoM), Center of Gravity (CoG), CoG of
  greater area, CoG weighed by the area, CoG
  weighed by the height, CoG of greater height,
  Minimum Point Criterium (MPC) weighed by the
  area, MPC weighed by the height, Average of the
  MPC, Average of the minimum and maximum
  MPC, MPC of greater area, and MPC of greater
  height.
- Operator of Comparison: Minimum t-norm.
3. A CONTROLLER IN SCD, FOR SIMULATION OF AN INDUSTRIAL GREENHOUSE

The greenhouses arise with the purpose of increasing the production, or simply to produce although the environmental conditions do not allow it. The objective is to obtain a suitable microclimate for the optimal development of the crops. With this idea we set out through SCD to simulate the process of climatic control in a greenhouse under a plantation of peppers by means of the management of the passive ventilation, fog-system and heat shield.

3.1 Input Variables

In order to do it, we have to create the controller previously and to this end we have considered like first step the designation of input variables that are, according to the experts [8], as follows:

♦ Solar Radiation: It allows the distinction between the night and the day since inner temperature and relative humidity of the greenhouse have considerable variations. For it, we can see two kinds of fuzzy sets: a) Night (function type L with a transition between the night and the day between 0 and 50 W/m²), and b) Day (function gamma type, with a slope of 50%, this fuzzy set really grows reaching the 100% truth from 50 W/m²).

♦ Temperature: In our crop of peppers, the experts considered necessary to settle down an optimal rank of temperature during the day of 23 to 27°C. During the night the optimal rank would be of 17 to 19°C. The defined fuzzy sets are: a) Minimum. b) Low nocturnal. c) High nocturnal. d) Low diurnal. e) High diurnal.

♦ Relative Humidity: As it happens to the temperature, the humidity is going to depend to a great extent on the type of crops and the phase of evolution of the same one. In this case the following fuzzy sets have been considered: a) Nocturnal minimum. b) Low nocturnal. c) High Nocturnal. d) Low diurnal. e) High Nocturnal.

♦ Wind Speed: It establishes a level of structural security of the greenhouse. It allows to determine in what moments the protection of the greenhouse prevails causing the total closing of the large windows due to strong winds. On the other hand it establishes, next to other variables, in what proportion to graduate the opening of the same ones when the conditions are not dangerous (gales). In one first approach three levels with respect to the wind speed could be considered: a) Soft. b) Moderate. c) Strong.

♦ Wind Direction: This allows to improve the air circulation in the interior of the greenhouse when the circumstances therefore require it, opening those windows that are in favour of the origin of the wind. The amount of opening will also depend on the speed of the same one. In the South of Spain the predominant winds are those of the West. The originating winds of the East are drier and warmer winds than the winds of the West and those of direction of the South and the North are practically null. This way it would be possible to said that the system of passive ventilation introduces and evacuates winds of the West and East through the lateral and superior large windows of the greenhouse. To this end the following fuzzy sets have been distinguished: the North, the South, the East and the West. Each one of these components takes a greater fuzzy membership degree when it indicates an exclusive direction.

3.2 Output Variables

Secondly the output variables settle down. For it is necessary to consider different aspects that we will study next. At times of high insulation the reduction of air temperature is normally carried out by means of the management of the ventilation. The most usual systems of ventilation are of passive type by means of lateral and superior windows. This takes to us to define the following variables of output:

♦ Superior Ventilation of the West and the East: it refers to the degree of opening (or closing) of the large windows located in the superior part of the greenhouse in favour of predominant winds of the West and the East.

♦ Lateral Ventilation of the West and the East: it refers to the lateral windows of the greenhouse.

The method of defuzzification chosen for these variables has been the Center of Gravity weighed by the Height. This method calculates the center of gravity of each one of the resulting fuzzy sets of the global set of output, receiving on the basis of its weight a different weight. It has resorted to this one method, because we think that the only different characteristic between the fuzzy sets of the type singleton is the weight that can take according to the activation degree of the control rules; when weighing by the weight the rules that activate in these fuzzy sets will take their degree from importance in the global result.

The following advance would be the implantation of systems of forced ventilation by means of motorized ventilators that make a better air flow when the farmer needs it and the environmental conditions do not cause it. But this has not been considered in the greenhouse as an object of our study.

Other variables of output are the following ones:

♦ Fog-System: Its fundamental objective is to increase the concentration of internal relative humidity of the
greenhouse. If this humidity is low, it rises through the water pulverization to high pressure. Indirectly, an increase of humidity also causes a small decrease of temperature. Two types of fuzzy sets are distinguished (singleton type) that indicate the state of operation, i.e. Off or On. In this particular case intermediate levels in the fog-system apparatuses cannot activate.

- **Heat shield**: It fulfills two functions depending on the time of the year. In winter, the unfolding of the same one aids the retention of temperature without power cost (heating). In summer, its extension absorbs part of the solar radiation to avoid an excessive elevation of the temperature in the interior. Similarly to the commented one in the fog-system two possible states through two diffuse sets would be distinguished: Off and On.

### 3.3 Base of Rules

Once we have defined the variables of input and output we go to the third step that is the construction of the Knowledge Base. The following groupings have settled down:

- **Safety Rules**: based on the speed and wind direction they determine the level of protection to apply on the structure of the greenhouse. An example of this is the following rule: *If Speed Wind is Moderate and Direction Wind is of the East then Superior Ventilation of East is Closed*.

- **Rules of Nocturnal Control**: They are going to be the ones in charge to establish the levels of opening of the different large windows next to the activation from the fog-system and heat shield in the nocturnal period. As example let's take the following rule: *If Solar Radiation is Night and Relative Humidity is Low Nocturnal and Temperature is Low Nocturnal and Speed Wind is not Strong then Superior Window West is Closed and Superior Window East is Closed and Lateral Window West is Closed and Lateral Window East is Closed and Fog system is Off and Heat shield is Off*. With this one rule the fog-system does not activate, because since the temperature is low would cause, if activated, a greater decrease, therefore all the greenhouse is closed to avoid losses of energy.

- **Rules of Diurnal Control**: They are going to manage the passive ventilation, the fog-system and the one of heat shield during the day. Example of a management of the temperature that fluctuates in greater measurement during the day that the humidity is: *If Solar Radiation is Day and Relative Humidity is High Diurnal and Temperature is Low Diurnal and Speed Wind is not Strong then Superior Window West is 50 and Superior Window East is 50 and Lateral Window East is Open and Lateral Window East is Open and Fog system is Off and Heat shield is Off*. By means of this rule one looks for to evacuate part of the humidity avoiding to lose much temperature.

The Knowledge Base is one of the most important parts of the system and is the one that more work costs to the experts. Nevertheless, the inclusion of Knowledge Base on the program is relatively fast and simple through the builder windows of antecedent (Figure 1) and consequent of each rule. These windows respectively show the user all the possible variables of input and output available next to the corresponding associated fuzzy sets to each one. In addition, in the formation of the antecedent there is the possibility of choosing between operators and/or between terms, as it is appraised in Figure 1.

### 3.4 Options of Inference Engine

Finally the Options of Inference Engine would settle down, in the present study have taken like operators the shown ones in Table 1.

The choice of the algebraic product as operator “and” has been exposed on the basis of the exposed thing in [13] where it reveals, through experiments, that the operator of the minimum does not work well like descriptor of the connective “and”.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implication</td>
<td>Minimum</td>
</tr>
<tr>
<td>Aggregation</td>
<td>Maximum</td>
</tr>
<tr>
<td>And</td>
<td>Algebraic Product</td>
</tr>
<tr>
<td>Or</td>
<td>Maximum</td>
</tr>
<tr>
<td>Comparison</td>
<td>Minimum</td>
</tr>
</tbody>
</table>

**Table 1: Inference Engine Options.**

### 4. SIMULATION IN SCD

Once the controller has been defined, we are going to make an analysis and exhibition of the results obtained through the simulation of the system in SCD. The simulation can be executed of three different forms.

#### 4.1 Simulation Step by Step

The computing process of the fuzzy knowledge based controller in SCD realizes five phases for obtaining the control actions. Such phases can be visualized through their corresponding windows. These phases are:

1. **Phase of Comparison**: This process compares the fuzzy sets. It compares the values of input (stumped according to the chosen fuzzification) and the different fuzzy sets that conform the linguistic labels of the input variable. With this we can weigh to what extent the fuzzificatons and the labels are superposed through the possibility concept (Poss).

2. **Phase of Activation of Rules**: With the values of possibility of each term, which form the antecedents in the knowledge base, the operators “and/or” and “not” are applied on these terms in each rule. With this the activation degree of each rule is obtained.
3. **Phase of Implication:** The implication process works on the output variables of the system. It applies the implication operator on each one of the fuzzy sets of each variable of output. Naturally this considers the degree of activation of the activated rules that influence this fuzzy set of output.

4. **Phase of Aggregation:** It is made through the aggregation operator. In this phase the resulting fuzzy sets of the phase of implication will be united. This is the phase that unifies the criteria expressed in all the rules.

5. **Phase of Defuzzification:** In the last phase the fuzzy information is processed (fuzzy set resulting of the phase of aggregation) and a concrete value (physical or real result) is obtained, for all output variables. As it has been said, there are several defuzzification methods that the system allows to use. In Figure 2, once we have selected the output variable, the resulting fuzzy set of the accumulation of calculations of Phase 4 will be visualized. Next to this, it is shown in a discontinuous outline the present value of output obtained in Phase 5. The system obtains it when it apply the defuzzification method to the mentioned fuzzy set of output.

4.2 **Computation of Output Variables for a Concrete Moment (or for a Concrete Value in the Inputs).**

The grouping of all the previously described phases gives rise to a computation global process. A fast form to provide the user the answer of the system to concrete input values, is by means of the window of computation of output variables.

4.3 **Simulation of Sequences of Input Values**

The possibility of evaluating the system at different moments from time (states) is going to provide a form to simulate the evolution in the control action before certain sequences of data.

This also allows us to register the behaviour of the system throughout the time, with the objective to make decisions on the system (to change the base of rules, or the definition of some labels...). This option is very important for the study of the system, its evaluation and its tuning (refining).

This evaluation is made through the window to compute output data sequences (Figure 3). This window appears with two marks of elements. On the one hand, in the superior part is the frame of input data which contains in its right part a listing with the input variables. Those that are checked will be visible in the graph of the left (sequence of input data). X-axis of that graph is the time, measured at "moments" in which each variable takes a particular value. Output variables (in inferior part) have a similar format.

In Figure 3 it can be seen how all the variables of input are constant except for the wind speed. An abrupt change in the wind speed generates an immediate closing...
of the windows. After that abrupt change, the windows return to open themselves. As the wind speed continues increasing slowly, there is a little while in which the windows return to close themselves. It can be seen that the rest of variables of output takes gradual values according to the wind speed that modifies its values of input at the different states of time.

5. CONCLUSIONS

We have presented here a fuzzy control simulation software of fuzzy control [5]. One of the most important characteristics is its flexibility, which allows to adapt to the possible necessities of any user because of its redefinition character. The interface allows, in a simple and pleasant form, to introduce in the basic principles on fuzzy controllers and thanks to the evaluation step by step, to know in a more detailed way the operation of a controller based on fuzzy knowledge. By these reasons we think that SCD could be a valid tool for didactic purposes.

The search of the implantation of the principles of fuzzy control in the agricultural production systems has been the great goal we look for with the elaboration of this software.

Fuzzy control would be one more tool that would allow farmers, scientists... to compile information and to obtain productive results. These productive results would be given by a “storage” capacity of the expert knowledge so that a “machine” could serve as support of aid to the decision.

The application of SCD to the climatic control of a greenhouse of peppers has allowed its simulation, so that we could obtain data to improve the crop.

It is in the intention in future work to implement physically the fuzzy controller. The objective would be not only the simulation of the system but also the real control of it on the behalf of SCD. For it we would have consequently to elaborate a system of data acquisition in real time of the input variables acting on the physical devices that manage the output variables. This way, a real vision of the controller would be obtained.

On the other hand the system can be extended including more functions, specially for the inference options (Table 1). Despite that, it is not so necessary. Perhaps it is more important to include new types of rules that allow us to express the knowledge base in a more comfortable way.

The presented work could also be used in the sense of approximation of functions. For example, let’s suppose in the scope of the agriculture that a function calculates the evapotranspiration of a crop that starts off from the time passed from its plantation. That function is complex to calculate mathematically. However, using the principles of fuzzy control, it could be feasible to calculate an approach to that function from simple rules.

REFERENCES

8. P. Lorenzo, Dpto. de Horticultura y Und. Fisiología del Centro de Investigación y Formación Agraria (C.I.F.A. – Almería) Junta de Andalucía, fisiobht@arrakis.es
MANUFACTURING SIMULATION
USING HIL SIMULATION TO SET UP AND TEST THE CONTROLLER OF A NEW TRY-OUT HYDRAULIC PRESS

Joseba Landaluze
Aron Pujana
Leire Maruri
Carlos F. Nicolás
Ana Martínez
IKERLAN Research Centre
E-20500 Arrasate/Mondragon. Spain
E-mail: jlandaluze@ikerlan.es

KEYWORDS
HIL (Hardware in the Loop), Virtual Prototyping, TRY-OUT hydraulic press, Real-Time simulation.

ABSTRACT
A new High Speed TRY-OUT hydraulic press has been designed and built by the ONA-PRES company in collaboration with IKERLAN. The press is capable of emulating the trajectory and operating characteristics of any mechanical press. The operating conditions are programmable and enable the dies to be set up under the same operating conditions they will have when they go into production in a mechanical press. In parallel to the press construction, a Virtual Prototype of the press was developed, so that the controller designed and built was set up and tested using Hardware-In-the-Loop (HIL) simulation with the Virtual Prototype. This paper presents the modelling and development process for the Virtual Prototype of the press, as well as some of the results obtained using HIL simulation, compared to those obtained with the real press.

INTRODUCTION
In overall terms there are two phases in the development process for a new mechatronic product: the design phase and the integration and validation phase. On the one hand, modelling has been traditionally used in the initial stages of the mechanical design phase, in order to obtain a parametric model or digital prototype. On the other hand, control-oriented modelling is used for the development of controllers. Frequently, both types of modelling, that for the mechanical design and that for the design of the control, are carried out completely independently.

Nowadays, the optimum development of new mechatronic products requires that the modelling and simulation cover all the stages of the design and integration phases, and also, that the mechatronic design and the design of the controllers be dealt with together. The link is the dynamic modelling and simulation.

The dynamic modelling for the design of the controllers is normally oriented to “off-line” simulation, that is to say, it is not Real-Time. However, the hardware and software tools available today make it possible to channel the modelling towards Real-Time simulation. Then, related to RT simulation, increasingly more everyday concepts appear, like Rapid Control Prototyping, Virtual Prototyping and Hardware-In-the-Loop (HIL) simulation (Lambert and Wöllermann 2000). All these are very interesting concepts from the Control Engineering point of view.

By means of the Rapid Control Prototyping, once the control algorithms have been designed, the controller can be tested with the physical plant without having the design for the final controller, using off-the-self hardware. Once the algorithms have been properly tested, the controller can be physically built. However, sometimes the real system to be controlled is not available, perhaps because the development has not finished, perhaps because it is very expensive and cannot be built just to test a component or some control functions. On other occasions the real plant is available, but the risk of damage that testing the controller could cause to the system is very high. The alternative is to test the controller with a RT model of the plant, that is to say, with a Virtual Prototype. This means that the controller has to see the Virtual Prototype as if it were the physical system to be controlled. So, the Virtual Prototype must behave in Real-Time like the plant would do in the dynamic functions and characteristics in which the controller intervenes. When a physical prototype of the controller, or even the final controller, is being used with the Virtual Prototype of the plant, this is called a HIL simulation.

HIL simulation-oriented Virtual Prototyping involves tough modelling demands for RT simulation, so the modelling tools to be used, as well as the platforms for the RT simulation must be chosen carefully. Some years ago, IKERLAN selected a methodology for developing Virtual Prototyping and HIL simulations (Otaduy et al. 2000), which has been applied in different projects (Landaluze et al. 2002a; Landaluze et al. 2002b; Fernández et al. 2003). More recently, and due to the increase in the number of alternatives available on the market, the development methodology and platforms have been reviewed and completed.
In 2003, the press manufacturer ONA-PRES, in collaboration with IKERLAN, developed a High Speed TRY-OUT hydraulic press. The press is capable of emulating the trajectory, speed and operating characteristics of any mechanical press. The operating conditions are programmable and enable the dies to be set up in the same operating conditions they will have when they go into production in a mechanical press. This means that the savings in time during set up and going into production are quite considerable. The TRY-OUT press developed, in addition to setting up dies, can be used as a conventional hydraulic press for production tasks. The press incorporates a high performance controller that basically controls the position/speed of the slide and its parallelism, as well as the hydraulic cushion the press is equipped with.

When the control algorithms were designed and the final industrial controller built, the mechanical-hydraulic part of the press still had not been completed. In any case, at the same time and as part of another project, a Virtual Prototype of the TRY-OUT press being built had been developed. This meant that the industrial controller was set up and tested in detail using HIL simulations. As a result, the integration process for the controller in the real press was then carried out in very little time. Figure 1 shows a photo of the TRY-OUT press built.

This paper presents the development process for the Virtual Prototype of the TRY-OUT hydraulic press and some of the results obtained in the HIL simulation, compared to the experimental results in the real press. Initially, the methodology selected for the development of the Virtual Prototype and for the HIL simulation is presented. The hydraulic press developed is then described. Subsequently, the different steps followed up to the construction of the Virtual Prototype and the HIL simulation environment are presented. Finally, some of the results obtained with the platform are given, compared to the results obtained with the real press.

DEVELOPMENT METHODOLOGY FOR HIL SIMULATION

IKERLAN’s Mechanical Engineering Department normally uses the Pro/Mechnica package for modelling and dynamically simulating mechatronic systems. Moreover, the Control Engineering Department uses SABER to model electronic and hydraulic systems, and to assess by means of simulation the control algorithms designed. Due to the experience available with these tools and the confidence in the results, the modelling and the results obtained from the simulation using those packages are used as a reference to validate the models carried out with other tools.

At IKERLAN an assessment was made of the commercial software tools available to automatically create applications. The focus was always on object-oriented modelling and the automatic handling of models, so that they constitute elements of a platform for Virtual prototyping and HIL simulation.

Figure 2 shows the framework finally selected and the development methodology for HIL simulation. According to this framework, firstly, object-oriented modelling is carried out with the Dymola or the 20-SIM packages. They were both selected mainly owing to the quality of the code generated and their ability to resolve algebraic loops. For the case of the hydraulic systems, like the TRY-OUT hydraulic press, Dymola/Modelica is used preferentially, due to the hydraulic libraries available and to the Modelica modelling language. The results of the simulations carried out are compared with the results of the simulations carried out previously with SABER (or Pro/Mechnica in other cases). This comparison stage is necessary because RT simulation-oriented modelling requires a simplification of models and a limitation of the bandwidths. Its effects have to be assessed and taken into account.

During a second stage, a submodel is created of just the parts to be included in a Virtual Prototype, removing unnecessary elements and those that will be part of physical components. In this way, the code is obtained via a Simulink S-Function.
The code for RT simulation is obtained from the overall model in Simulink with the RTW software (Real Time Workshop) from MathWorks, Inc. Simulink is used as an integration environment, creating a model of everything included in the Virtual Prototype, adapting the inputs and outputs in accordance with the emulation hardware, and bearing in mind the interface between the Virtual Prototype and the real elements. Although at IKERLAN until recently dSPACE hardware was used for RT simulation, very good results have been obtained lately with xPCTarget, using PC/104 hardware similar to xPCTargetBox. Therefore, both hardware platforms have been validated and adopted.

For RT simulation the step size used is very important. It has to be in line with the sampling rates of the physical elements included in the HIL simulation. Very restrictive demands in terms of sampling times and bandwidths lead to a process of simplification of models in Dymola/Modelica, until the RT simulation can be run with the integration step required. Obviously, the hardware of the platform on which the simulation is carried out is a determining factor in the limitation of sampling times.

With the dSPACE platform, the ControlDesk tool is used to create user panels and to control the development of the RT simulation. Moreover, the MotionDesk package is used for the 3-D animation of the mechatronic system in a virtual world. With the xPCTarget platform, the Simulink environment is used to create the basic panels from which to monitor the interesting variables and control the simulations. In addition, the Virtual Reality Toolbox is used to create the virtual world and the 3-D animation.

In short, from the point of view of the user interface to monitor and control the RT simulation, similar results were obtained in the two platforms, with the tools mentioned.

**DESCRIPTION OF THE TRY-OUT HYDRAULIC PRESS**

**Description of the whole system**

The new TRY-OUT hydraulic press developed by ONAPRES in collaboration with IKERLAN is a press that can operate as a conventional hydraulic press in production operations and as a TRY-OUT press emulating the operating conditions of any mechanical press (González et al. 2003). Functioning as a High Speed TRY-OUT press, it can be used to set up dies under the same operating conditions that they will have when they go into production in a mechanical press. The trajectory of the mechanical press to be emulated is introduced from the user interface and the TRY-OUT press is capable of carrying it out. To do this, the press slide is moved by four cylinders, with each cylinder controlled by a servovalve that during the emulation phase injects oil from a high-pressure accumulator unit. In this way, each cylinder is controlled individually, so that the slide carries out the movement of the programmed curve and controls the parallelism during the drawing tasks, achieving the high speeds of mechanical presses, more than ten times the drawing speeds of the press operating as a conventional hydraulic press.

Figure 3 shows a simplified general diagram of the press. It basically consists of a slide (1) driven by four hydraulic cylinders (2). It has a pump unit (3), a conventional hydraulic control unit (4) and an oil tank (7). Each cylinder is controlled by a servovalve (5), which uses the oil from a high-pressure accumulator unit (6), or if not is controlled by the conventional control unit (4). The press also has a controlled hydraulic cushion (9), driven by four cylinders that are controlled individually by means of four
servovalves. During the whole drawing process, the slide works in conjunction with the controlled hydraulic cushion, which enables the optimisation of the drawing force and the carrying out of the force profiles used in the mechanical press to be emulated.

Functioning as a conventional hydraulic press, the servovalves (5) do not come into operation and the press carries out the operating cycle controlled by the conventional hydraulic control unit, which is controlled by the press PLC. Figure 4a shows a typical cycle of the press operating as a conventional press. It shows the positions of the slide and the hydraulic cushion. The slide comes down from the top position ha to the programmed point hb at constant fast speed. It then goes down to the bottom hm at constant operating speed. From that bottom position, it goes up to the top, completing the cycle. During the downward phase at drawing speed and before the drawing starts, the slide reaches the sheet metal (10) and the holder plates (11) of the controlled hydraulic cushion, and push them to the bottom point hm.

Functioning as a High Speed TRY-OUT press, the press carries out a cycle like that shown in Figure 4b. Initially the slide comes down from the top position ha to the programmable point hb at fast speed, just as if it were a conventional hydraulic press cycle. From that point, the slide brakes until it reaches the programmable height hc. During this braking phase, there may be the contact between the slide and the sheet metal (10) and the holder plates (11) of the controlled hydraulic cushion. From the hc point, the servovalves (5) come into operation and control the flow in the upper chambers of the slide cylinders, so that the slide moves in the required emulation trajectory until the bottom point hm. The flow is provided by the high-pressure accumulator unit (6). At the slide’s bottom point hm, the servovalves (5) stop operating and pass control to the conventional hydraulic control unit, which completes the cycle as if it were a conventional hydraulic press.

During the emulation phase, between hc and hm, the slide is controlled by the specially developed controller, which carries out position control and parallelism control. Moreover, the same controller controls the hydraulic cushion, implementing the position and force control algorithms required (Landaluze et al. 2002a).

Figure 5: Schema of the Slide Control Algorithm

Figure 6: Modelling of whole TRY-OUT Press in SABER

Description of the controller

The TRY-OUT press controller controls the position and parallelism of the slide and the position/force of the hydraulic cushion. It receives information about all the pressures of the two chambers in the cylinders and their positions, and generates the set-point values for all the servovalves. Moreover, it communicates with the press PLC by means of digital signals, to synchronise the different work phases and operating states. To control the slide, a diagram like the one shown in Figure 5 is implemented. The control block (40) generates a feedforward set-point value for each servovalve based on the theoretical speed $v^*$ of the curve to be emulated by the cylinder, the pressures $p_1$ and $p_2$ in the upper and lower chambers, the areas $s_1$ and $s_2$ of both chambers and the approximate weight $w$ of the slide, in accordance with the formulas:

$$p_2 = \frac{p_1 \cdot s_1 - w \cdot s_2}{s_2}$$

$$i_f = \frac{10 \cdot s_2 \cdot v^*}{A_0 \cdot C_d \sqrt{\frac{2(p_s - p_2)}{p}}}$$

in which $A_0$ is the maximum flow area of the control servovalve, $C_d$ is the discharge coefficient, $\rho$ is the mass density of the fluid and $p_s$ is the inlet pressure of the oil into the servovalve. The output of the control block (40) is $i_f$. The control block (41) controls the position in closed loop. The block (42) controls the parallelism of the slide. From the slide positions measured, a position reference $p_{ref}$ is obtained for each cylinder, which is compared with the directly measured or estimated position.

The industrial controller built consists of two parts: the controller itself, built with Siemens SICOMP SMP-16 hardware using the RTOS RMOs3, and the Configuration and Monitoring Unit (user interface). The user interface is in the computer on the press and communicates with the controller by means of the Profinet bus. Figure 12 shows on the right both parts of the industrial controller.

VIRTUAL PROTOTYPE OF THE TRY-OUT PRESS

This section presents the modelling and simulation steps followed up to the building of the Virtual Prototype and the carrying out of the HIL simulations.
Modelling and simulation in SABER

As mentioned above, the simulations using the SABER package are used for the design and assessment of control algorithms and as a reference for the modelling carried out in Dymola/Modelica and in Simulink during the process of developing a Virtual Prototype. Although the tuning of the controllers in simulation was done with partial models of the slide and hydraulic cushion, a model of the complete system was finally made. Figure 6 shows the complete model of the TRY-OUT press. Only half of the press has been taken into account, the front part, with the rear part being taken as symmetrical. The same hypothesis was maintained for the construction of the Virtual Prototype. The model has two cylinders and two TN63 servovalves in the slide part and another two cylinders with their corresponding servovalves in the cushion part.

The standard control block implements the conventional hydraulic control of the slide, which is used during the fast downward and fast upward phases, and is modelled by means of ON-OFF valves. There is another block for the press controller, which implements the slide position and parallelism control algorithms, as well as the cushion position/force algorithms. Another block generates set-point values for all the valves in all the operating phases.

As an example of the results of the simulations, Figure 7 shows a complete cycle of the TRY-OUT press, emulating during the emulation phase the trajectory of a sinusoidal press at a rate of 10 spm. The figure shows the positions of the slide and the cushion. In the cushion a pre-acceleration of 90 mm has been taken into account. Figure 8 shows the emulation phase in more detail. At the top the positions are and at the bottom the speed set-point value and the slide speed. As can be seen, after the braking stage the slide accelerates in a controlled way until it achieves the speed curve it has to emulate. In this simulation, it has been supposed that the impact between the slide and the cushion occurs once the emulation has started. It can be observed that this impact disturbs the position of the slide, as can be seen in the speed graph, although it rapidly recovers. By way of conclusion, and for the impact to have the least possible effect on the drawing process, it was seen that it was more interesting for the impact to take place during the braking process or whilst the slide is accelerating. In any case, all this can be programmed on the press developed.

Modelling and simulation in Dymola/Modelica

Dymola/Modelica allows technological modelling of multidisciplinary systems. Models of the Dymola environment fulfil all object-oriented modelling specifications. Assumed symbolic procedures aimed to create efficient model descriptions are also used for code generation. The largest part of the model is written in ODE format, whereas inline iterative methods are used to solve large or non-linear algebraic loops.

The first step in the process for building a Virtual Prototype is the modelling and simulation of the complete system in Dymola/Modelica. In the IKERLAN case, this modelling is verified by comparing the results with those obtained with the SABER package, which is used as a reference. Once the results are acceptable only the part to be integrated into the Virtual Prototype is left. Figure 9 shows the continuous model of the mechanical-hydraulic part of the TRY-OUT press done in Dymola/Modelica (v. 5.0).

Just as with the modelling done with SABER, the slide model consists of two double chambers (c1A and c1B). The housing of the cylinders is taken to be fixed to the structure, whilst the piston located in the lower chamber moves sustaining the weight of the slide. An approximate weight of 30 t per cylinder was taken into account.
For the slide to carry out the movements typical of hydraulic presses, the ON-OFF valves that appear in the model are used. Depending on the oil passage area taken into account in each valve the different upward or downward speeds are achieved. For a rapid descent, at a constant speed of about 300 mm/s, valve B3 is activated, extracting oil from the lower chambers. For the slide to descend at a much slower speed, as in the case of drawing with a conventional press, the valve B2 is activated, which has a much smaller oil passage diameter, as well as the flow-sources Q1 and Q2. The upward movement is achieved by activating valves B1, B5A and B5B. B1 injects oil into the lower chambers and the valves B5 extract it from the upper chambers. In this way, an upward speed of about 270 mm/s is achieved.

To bring the slide down in a controlled way during the emulation phase, the TN63 servovalves are used backed by the ON-OFF valve B2. The servovalves introduce flow into the upper chambers, whilst the action of B2 serves to maintain constant the pressure of the lower chambers for the duration of the emulation phase.

In the part of the model corresponding to the cushion, the cylinder housings are fixed to the bottom structure, whilst the pistons, located in the upper chambers, move supporting the weight of the cushion and the die. Each cylinder supports a corresponding part of the weight, as well as the friction forces. The position/force of the cushion is controlled by the Rexroth 4WRDE32V-600L servovalves, which extract or inject flow into the lower chambers. The pressure in the upper chambers is maintained constant at all times.

Real-Time modelling of the TRY-OUT press

The modelling for the RT simulation is obtained in Simulink. On the basis of the model in Dymola an S-Function is created which is integrated in Simulink. The model is completed with the blocks that represent the adaptation of the inputs and outputs of the model to those of the RT simulation hardware. Figure 10 shows the model done in Simulink for the case of the TRY-OUT press.

In the model shown in the figure, the basic part is the S-Function represented by the “TRY-OUT press” block. In addition, there is the “PLC” block, which implements the logic followed by the press PLC based on the inputs it receives, from both the press control panel and the industrial controller. The rest are blocks that cable the physical inputs and outputs to the inputs and outputs of the “TRY-OUT” and “PLC” blocks mentioned.

Schema of the Virtual Prototyping for HIL simulation

The Virtual Prototype of the TRY-OUT press consists of three components:

1. The Virtual Prototype itself of the mechanical-hydraulic part of the press, including the press PLC, which is the RT simulation. The code is run on the dSPACE platform, on a DS1103 card, or on hardware compatible with xPCTarget from MathWorks.
2. The real industrial controller for the TRY-OUT press, which controls the slide and the hydraulic cushion.
3. The interface hardware, which on the one hand adapts the RT simulation hardware TTL signals to the industrial level required by the controller, and on the other transforms the encoder signals to SSI format, so that the controller can read them properly. It also implements the press control panel.

Both part 2 and part 3 make up the physical elements that interact with the RT simulation (component 1), so they are HIL elements. The three blocks mentioned appear in the diagram in Figure 11. The diagram shows the case in which the platform for the RT simulation is the xPCTarget compatible hardware, but it would be similar for the other platform chosen, with the dSPACE DS1103 card.

Figure 12 shows the complete Virtual Prototype on the dSPACE platform. The computer with the dSPACE card is on the left. The computer has been fitted with two screens, so that different things can be monitored on each. One screen shows the control panels carried out with ControlDesk, which monitor the status of the simulation.
and the values and graphs for the most interesting variables. The other screen shows the 3-D animation in Real-Time of a virtual press, done in MotionDesk. The interface hardware incorporating the press control panel can be seen in the middle of the Virtual Prototype. The industrial controller of the press is on the right, made up of the controller itself and the Configuration and Monitoring Unit.

On the dSPACE platform, the complete simulation of the system, including inputs and outputs, is carried out in 3 ms, whilst the industrial controller closes all its loops in 1 ms.

For the simulation on the xPCTarget platform, PIP8 hardware from the MPL firm was used, which is similar to xPCTargetBox from MathWorks. It is based on a Pentium III CPU at 700 MHz. For the construction of the Virtual Prototype of the Try-Out press, as a certain number of analogue and digital inputs and outputs are required, the system was completed with Diamond-MM-32-AT and ONYX-MM-XT boards, both from Diamond Systems. On this platform, Simulink was used to monitor and control the RT simulation and the Virtual Reality Toolbox from MathWorks for the 3-D animation. With this platform, the RT simulation of the complete system is performed in 700 µs, a time much more in line with the sampling time of the real controller.

Obviously, due to the differences in the simulation step sizes with the two platforms chosen, the results are slightly different, with the simulation on the xPCTarget platform generally proving to be much more stable.

RESULTS

The Virtual Prototype of the Try-Out press was used to set up and test the controller developed. On the one hand, the control algorithms were tuned up and, on the other, all the operating conditions and all the possible cases were tested. This is one of the great advantages of Virtual Prototyping, which makes it possible to simulate all the test cases imaginable, even the most unlikely scenarios, and without running any risk of damaging the physical system. Once that the Try-Out press had been built, it was necessary to fine-tune the control algorithms again, but the orders of magnitude for all the parameters were the same. This meant that the integration and set up in the real press was carried out in a very short time.

Now some examples of the results obtained with the Virtual Prototype and experimentally on the real press are presented. Figure 4b shows the positions of the cushion and the slide on the real press, when the Try-Out press was emulating during the simulation phase the trajectory of a FAGOR press at a rate of 16 spm. The emulation phase is 240 mm, of which 80 mm are used for acceleration and 160 mm for actual emulation. Figure 13 shows the same case obtained with the Virtual Prototype on the dSPACE platform. As can be seen, the braking phase is carried out with a gradient on the actual press. On the Virtual Prototype braking is much faster. When the slide reaches the bottom point hm and before the controller passes control to the press PLC (conventional controller), there is a "decompression" phase, in which the pressure in the upper chambers of the slide cylinders is close to zero, to facilitate the slide going up. As can be seen in the graph obtained with the real press (Figure 4b), this decompression phase takes time, during which the slide remains at the bottom position hm. In the Virtual Prototype (Figure 13), the decompression phase is almost instantaneous.

Figures 14a and 14b show the speed set-point values and the slide speed of the press during the simulation phase. Figure 14a shows the experimental values on the real press, when emulating the trajectory of the FAGOR press at a rate of 16 spm. Figure 14b shows the values obtained with the Virtual Prototype and HIL simulation. It can be seen that the results are similar, although in the real case it takes the slide longer to accelerate.

Figures 15a and 15b show the force exercised by the cushion after the impact with the slide, during the drawing phase. Figure 15a shows the result on the physical press, when emulating the trajectory of the FAGOR press at a rate of 4 spm. Figure 15b gives the data obtained with the Virtual Prototype. The force set-point value is 500 kN per cylinder. As can be seen the results are very similar. There is a difference at the end of the drawing, due to the already explained "decompression" effect. On the press built, to facilitate that phase, a controlled reduction of the force is carried out at the end of the drawing, which does not occur on the Virtual Prototype.
CONCLUSIONS

More and more object-based modelling tools with high quality code generation options are appearing onto the market. This, together with the new hardware available, now make technologies like Rapid Control Prototyping and Virtual Prototyping possible for HIL simulation. All this means that the development time for controllers is greatly reduced and that controllers can be set up and tested with Virtual Prototypes of the system to be controlled. It is possible to test all the design specifications and all the operating conditions without fear of damaging the real plant. These technologies that are being extensively applied in the automotive world can be applied as well in the design process for any mechatronic product. Ikerlan has recently chosen a development framework for Virtual Prototyping and HIL simulation, which is presented in this paper. Moreover, the paper is completed with an application case, where the development of the Virtual Prototype for a High Speed TRY-OUT hydraulic press is explained. The Virtual Prototype was used to set up and test the industrial controller designed for that press, when the press had still not been finished. This led to a great reduction in the time required to integrate and set up the controller in the physical press, once it was built.

ACKNOWLEDGMENT

The material of this paper has been backed in part by the Spanish Ministry of Science and Technology and FEDER (research project DPI2002-04438-C02-01).

REFERENCES

THE SIMULATION OF POLISHING PROCESSES AS BASIS FOR DESIGNING NEW GRINDING TOOLS

Wojciech Kacalak  
Stławomir Makuch  
Błażej Balasz  
Robert Cincio  
Mechanical Engineering Department  
Technical University of Koszalin  
75 – 620, Koszalin, Poland  
E-mail: kacalakw@tu.koszalin.pl  
E-mail: makuch-s@wp.pl

KEYWORDS
Abrasive machining, polishing, flexible grinding tools

ABSTRACT
The modelling of working processes is a very important scientific problem. This scientific problem takes special significance, when it is applied to abrasive machining, particularly polishing with flexible grinding tools. In the grinding zone – during the polishing process – even more random variable phenomenon appear than in other ways of abrasive machining. This random variable of polishing process as head factor is decisive for the need of modelling the polishing process. This article presents the complex model of the polishing process and the characteristics of the simulation system in the polishing process. This article presents the analysis of microcutting processes with grains located on flexible carrier. This process includes an additional typical quality of flexible grinding tools which is the change of mutual location of grains during polishing working. Results of mathematical simulation of burden blade and forming process of the worked surface are introduced in this article. The new conceptions of designed grinding tools are introduced in this article too.

SPECIFICATION OF THE POLISHING PROCESS
Polishing finds wide applications as an important technological method of superficial processing. Its main aim is to decrease roughness as well as conferment of gloss of the worked surface. Growing requirements relating to the geometrical structure of surface in the aspect of exploational property induce the search of new methods as well as improvement of the known and applied ones so far (Lu et al. 2002). Essential progress in polishing was obtained after the use of flexible grinding tools. Fixing the grain on flexible base changed the interaction on the workpiece in an essential way. It caused many after – effects, namely:
- it cushions percussive blow of grains against the workpiece contributing thus to the increase of the durability of grain,
- it decreases the intensity of heat flux, which considerably limits the possibility of thermal surface flaw,
- it disposes the possibility of creating microcracks in brittle materials,
- it minimalizes diversification of grain height thanks to which it enables the increase of the number of active grains,
- it decreases the speed of covering the active surface tool,
- it increases interaction contact time of grinding grains with working surfaces as opposed to rigid fixed tools,

The polishing tools with porous flexible bond are characterized by large stability of properties during their use. This quality of flexible grinding wheels allows for the mechanization and automation of polishing process as well as enables easy control for obtaining specific effects of the polishing process (Kacalak and Makuch 2003).
The working flexible grinding tools showed many advantages. They may be used for processing strangely shaped surfaces. The flexibility of tools enables makes the shaping of complex surfaces convenient.

THE STRUCTURE OF A COMPLEX MODEL OF THE POLISHING PROCESS
In the polishing process besides microcutting processes, scratching surfaces, cut down inequalities, friction plays an important part as well as physical–chemical phenomena which are connected with it (Kacalak and Pluta 1983). The participation of phenomena evoked by mechanical and molecular co–operation and heat emission depends on many factors, connected with stereometry of contact friction bodies as well as their physical properties. In the polishing process the part of microcutting processes in processes connected with removing material from the working surface is relatively large. It depends to a considerable extent on the size of geometry, fixed way and position of grinding grains in grinding wheel. 
Depending on these factors cutting, scratching surfaces with large occurrence of plastic deformation, or only elastic and plastic deformation of the workpiece may take place. Irrespective of the mentioned processes, during polishing
a workpiece abrasion of oxide films forming on the surface occurs as well as smearing over of vertex inequalities which are heated to semi–liquid state on the polished workpiece.

The fixed stiffness of grinding grains in the grinding wheel has essential influence on the contribution of the cutting process in the removal of the workpiece. The more stiff the grinding grains are fixed, the larger the contribution. The target of the polishing process is the removal of inequalities and polishing as well as giving a gloss to the surface. That is why the contribution of the cutting process in the polishing process shouldn’t be too large, because then phenomena, which are connected with the friction process, from which the gloss of the workpiece depends, have a small part. From the other side, the part of the cutting process can’t be too small because it limits the efficiency of the process and requires small roughness of the surface before polishing.

We may come to the conclusion that for the operation of preliminary polishing of large efficiency we should use polishing tools of relatively small flexibility, while for final polishing we should use wheels of large flexibility of grain fixation. Polishing with flexible grinding tools is an intermediate process between the grinding process and the polishing process, with wheels which are impregnation with abrasive compounds.

SIMULATION SYSTEM AND ITS CHARACTERISTICS

In the first stage we defined the initial parameters of the polishing process i.e. the depth of caving grinding wheel in workpiece, velocity of grinding tool, velocity of workpiece, number of transition grinding tool as well as the method of the polishing process (concurrent or backward).

The next step of simulation is generation of grinding grains. The shape of every grain is described in the form of a matrix thanks to which it is possible to modify the shape of grain during the simulation process, caused by the interaction of the grinding grains and the workpiece. Every grain is treated individually, that makes analysis of grain burdens possible, calculation of the cutting forces per each grain, interaction forces between grains as well as elastic deflections of single grinding grains. Depending on the grain size, the size matrix description of the grinding grain is selected and additionally the grain is located at random in the limits of the matrix, which prevents the formation of too regular grinding wheel structure (Balázs et al. 2001). The next step of simulation is generation of flexible grinding wheel. The rise of grinding wheel arises due to the composition of generated grain matrix in to one large grinding wheel matrix, between the grains matrix set type lines are inserted which characterize the bond. The matrix record of all grinding wheel and individual grinding grains enables the simulation of elastic displacement grains in any direction (elastic deflections, rotations) through operations on matrix (Balázs and Królikowski 2002).

The generation of grinding wheel surface and workpiece surface are followed by simulation of the polishing process. During the simulation process parameters of individual grains are monitored as well as the state of the workpiece and the state of entire the grinding wheel. On the basis of the measured values current parameters of the simulation process are calculated: instantaneous cutting force, as per each individual grinding grain, roughness of surface. Subsequent to the simulation process additional information is obtained about all grinding wheel state and workpiece state (roughness, waviness).

![Figure 1: Diagram of simulation system](image)

THE ANALYSIS OF THE MICROCUTTING PROCESS WITH FLEXIBLY FIXED BLADES

In real conditions, in grinding wheels with the grains of large flexibility, the cutting blades in the cutting zone undergo linear and angular displacement under the influence of cutting resistance. The values of displacement of the cutting blade, which is in contact with the workpiece, depend on its position in the grinding wheel, working conditions, geometrical and physical properties of working surfaces, flexibility of the grinding wheel and burden state blades which are side by side as well as
interaction between grains which are side by side. The
displacement is different for the different blades and they
shall be variable along the working place of the blade. In
Figure 2 it presents polishing surface process with flexible
grinding tools with considering typical quality of flexible
grinding tools which is the change of mutual location of
grains during polishing working. The change of mutual
location of grains depends on local flexible grinding tools
as well as the condition of deepening of caving blades in
workpiece and this quality is one of the important factors
which have influence on the efficiency and results of
grinding processes. The acquaintance of the mechanisms
of polishing surfaces is especially important at the
selection of processing parameters in order to increase
efficiency of polishing surfaces with grinding grains which
are located on flexible carrier (Kacalak and Makuch 2003).

The equilibrium equations for each grain are described by
the following relationship:

\[ P_S - Q_V + \Sigma P_O = 0 \]

Where:

- \( P_S \) – interaction force bond on grinding grains [N];
- \( Q_V \) – normal component cutting resistance [N];
- \( P_O \) – interaction force between grinding grains [N];

Figure 3: Distribution of forces interacting with grinding
grains

The following system of equations result from figure 3:

\[
\begin{align*}
P_{S1} - Q_{V1} + P_{O12} - P_{O14} &= 0 \quad \text{(Grain Z1)} \\
P_{S2} - Q_{V2} - P_{O21} + P_{O23} - P_{O25} &= 0 \quad \text{(Grain Z2)} \\
P_{S3} - Q_{V3} - P_{O32} - P_{O36} &= 0 \quad \text{(Grain Z3)} \\
P_{S4} - Q_{V4} + P_{O45} + P_{O41} - P_{O47} &= 0 \quad \text{(Grain Z4)} \\
P_{S5} - Q_{V5} - P_{O54} + P_{O56} + P_{O52} - P_{O58} &= 0 \quad \text{(Grain Z5)} \\
P_{S6} - Q_{V6} - P_{O65} + P_{O63} - P_{O69} &= 0 \quad \text{(Grain Z6)} \\
P_{S7} - Q_{V7} + P_{O78} + P_{O74} &= 0 \quad \text{(Grain Z7)} \\
P_{S8} - Q_{V8} - P_{O87} + P_{O89} + P_{O85} &= 0 \quad \text{(Grain Z8)} \\
P_{S9} - Q_{V9} - P_{O98} + P_{O96} &= 0 \quad \text{(Grain Z9)}
\end{align*}
\]

After defining all the data occurring in equilibrium
equations and making mathematical transformations a
system of equations was obtained which describes the
depth of flexible grinding tools in workpiece.

By solving this system of equations for subsequent points
in the working zone a blade grinding grain trajectory was
indicated which was located on flexible carrier.
Trajectories of subsequent transitions of the polishing
process with flexible grinding tools were modelled
considering the effects of previous transitions.

Figures 4, 5, 6 present the results of mathematical
simulation for blade burdens without considering
interaction force between grains, when the value of
interaction force between grains is relatively small (it is
close to value of resilience force fixing grinding grains),
and when the value of interaction force between grains is
relatively large (much bigger than the value of resilience
force fixing grinding grains). Figure 8 presents the
variation of roughness maximal height trajectory generated
by flexible grinding grains \( R_p \). The principal for defining
this parameter is presented in Figure 7.
Figure 4: Shape of smoothened surface with flexible grinding grains without considering the interaction force between grains.

Figure 5: Shape of smoothened surface with flexible grinding grains when the value of interaction force between grains is relatively small.

Figure 6: Shape of smoothened surface with flexible grinding grains when the value of interaction force between grains is relatively large.

Figure 7: Indication of $R_m$ parameter.

Figure 8: Value of $R_m$ parameter.

The worked out method of modelling the polishing process with flexible tools with consideration of interaction force between grains makes it possible to obtain information about mechanisms of polishing surfaces and behaviour of grains during polishing working. Together with enlarging the value of interaction force between grains, the height of inequality in individual vertical cross-sections (blade trajectory movement) decreases but the differences in shape locations in these vertical cross-sections are considerable. The decrease of interaction force causes levelling of inequalities and level of shape location in individual vertical cross-sections. By applying this method the principles of selection parameters of the polishing process in order to increase the efficiency of the polishing process may be specified.

**CONCEPTS OF NEW GRINDING TOOLS FOR POLISHING**

Regular or similar to regular structure is produced in order to obtain favourable exploational properties or in order to achieve decorative effects. The exploational effects of surface of regular structure depend on their application conditions. Surfaces of regular geometrical structure in the form of cavities or grooves can have applications as the surface of flat or rotation guides, resistance surface instrument and handle surface instruments as well as gauge surface instruments.

The results of research simulation polishing process can be used for the new method of forming discontinuous active surface flexible grinding grains. These tools shall be used in forming the surface workpiece of regular structure.
Figure 9 presents various kinds of discontinuity of active surface flexible grinding tool as the aggregates with grinding grain form which are located regular or irregular as well as the layer of grinding grains of increased stiffness (impregnated bands).

Fig. 9 Shape of active surface flexible grinding tool of discontinuous surface grinding wheel: a) grinding aggregates located in an irregular way, b) grinding aggregates located a regular way, c) band of grinding grain of increased stiffness of bond (impregnants).

The superiority of a flexible tool over a stiff tool in the smoothening process and other ways of surface working occurs in weakening the influence of different grain height on workpiece roughness. The most protruding grinding grains are pressed stronger into flexible bond and as a consequence their value depth of caving in workpiece gets smaller, and the roughness of workpiece decreases. The roughness of surface polished with flexible grinding tools, to a considerable degree, depends on the intensity of removing inequalities of the initial surface, and this intensity is conditioned with the quantity of grains which are in contact with the workpiece.

REFERENCES


BIOGRAPHY

WOJCIECH KACALAK was born in Zduńska Wola, Poland. He studied Mechanical Engineering at the Technical University of Łódź, where in 1970 he obtained his master degree, in 1974 doctor’s degree and since 1989 professor of technical sciences. Since 1970 he has been working at the Mechanical Faculty of the Technical University of Koszalin doing research on optimization of manufacturing processes with application of neural networks and fuzzy logic.

SLAWOMIR MAKUCH was born in Koszalin, Poland. He studied Mechanical Engineering at the Technical University of Koszalin, where in 2001 he obtained his master degree. He is a Ph.D. student at the Faculty of Mechanical Engineering of the Technical University of Koszalin doing research on computer simulation and optimization of grinding processes.

BLAŻEJ BALASZ was born in Trzcianka, Poland. He studied Mechanical Engineering at the Technical University of Koszalin, were in 1994 he obtained his master degree, in 2003 doctor’s degree of technical sciences. Since 1995 he has been working at the Mechanical Faculty of the Technical University of Koszalin doing research on computer simulation of grinding processes.

ROBERT CINCIO was born in Gorzów Wielkopolski, Poland. He studied Applications of Computers in Engineering at the Technical University of Koszalin, were in 2002 he obtained his master degree. He is a Ph.D student at the Faculty of Mechanical Engineering of the Technical University of Koszalin doing research on computer simulation and optimization of grinding processes.
MODELING AND ANALYSIS OF LOCAL ENERGY VARIATION IN GRINDING ZONE IN THE PERSPECTIVE OF THE SURFACE AREA QUALITY

Tomasz Królikowski
Błażej Balasz
Tomasz Szatkiewicz
Wojciech Kacalak
Technical University of Koszalin
Department of Fine Mechanics
75-620 ul. Raclawicka 15-17 Koszalin, Poland
e-mail: krolikow@tu.koszalin.pl

KEYWORDS
Grinding process, modeling.

ABSTRACT
The research results presented in this paper have been based on simulative research conducted for generation surface grinding and parameters most commonly applied at present for grinding wheels with a continuous and discontinuous active flank (Balasz 2001, Królikowski 2002, Balasz2002, Królikowski 2001). The paper presents results processed with the help of a computer simulation method (empirically verified), components of the grain work in the zone of the contact between the grinding wheel and the object being ground. Apart from this, the results of research into the influence of the non-continuity of the active flank of the grinding wheel on the change of the process’s energetic parameters have been presented.

ENERGY OF GRINDING WITH GRINDING WHEELS WITH CONTINUOUS AND DISCONTINUOUS ACTIVE FLANK

The simulation results presented in this paper prove the fact that the grinding process, and work connected with this, occur still outside point po, that is beyond the plane which goes through the grinding wheel axis (Fig.1). For the case presented and for the kinematic conditions assumed: \(v_f=5\text{m/min}, v_s=120\text{m/s}, a_e=0.02\text{mm}, \) this point (pk) is 0.45mm away from po. It is evident from the diagram presenting the distribution of grinding energy that during grinding with a grinding wheel with a discontinuous active flank the proper work per one abrasive grain is smaller than for a continuous surface of the grinding wheel, and the zone of active machining of grains is longer (Fig.3). The energy distribution in the initial range of lengths of the zone from 0 to 0.2mm (Fig.2), serves to prove the fact that the impact of grains upon the ground surface is different for continuous and discontinuous active flanks. In the final range of the grinding zone from 2mm to 2.45mm from initial point pp=0, presented in enlargement in Fig. 3, it is unambiguously evident that the grinding wheel grains with a discontinuous active flank do a lot of more machining work than grains in a grinding wheel with a continuous active flank.

The simulation of a machining process allows for the determination of variations in work carried out by grains located in a defined elementary strip of the active flank of the grinding wheel in the machining zone and in a defined duration of this process, i.e. ti. This is presented in Fig. 4 (variations in machining work for time ti=0.102 ms, with grains located in the zone of contact with the object machined). In this picture, one can also define the number of grains which are in contact at the same time as well as their loads connected with machining work carried out. The figures 4 clearly reveal an influence of the macro-geometry of the active flank of the grinding wheel on the load of grains in the grinding zone, which is distinctly larger for grains of a grinding wheel with a discontinuous active flank (NC) (Fig.4b). Active grains of a grain with a continuous active flank (C) in the machining zone are less loaded, they perform less machining work, while the number of grains which perform machining simultaneously is larger (Fig. 4a).
Fig. 1. Variability in the grinding zone: work $E_r$ of machining with individual grains, work $E_{sw}$ of scratch creation by individual grains, work $E_{sup}$ of creation of fashes, proper work $e$, for grinding wheels with continuous (C) and discontinuous active flank (NC)

Fig. 2. Distribution of work in the grinding zone: a) start of machining zone, b) end of machining zone

Fig. 3. Variability of energetic parameters, zone 1-output zone 2- final part of the grinding zone outside point $p_0$

Designations in the diagrams determine the following:

- $E_r$ - work (energy) of grinding per one abrasive grain,
- $(E_{sw,NC}, E_{sw,C})$ - work of chip creation with grains in a grinding wheel with a discontinuous (NC) and continuous (C) active flank,
- $(E_{sup,NC}, E_{sup,C})$ - creation work of a fash and plastic strains of a material for grinding wheels with a
discontinuous (NC) and continuous (C) active flank,

- \((c_{NC}, c_{C})\) - proper work of machining with one grain with a discontinuous (NC) and continuous (C) active flank.

Fig. 4. Variation in local values of micro-machining work with grains in the grinding zone while machining with a grinding wheel with a: continuous active flank (C) b: discontinuous active flank (NC) for time \(t = 0.102\) ms

The load of grains in the grinding zone is characterized by a large variability along the path of their work. At an analysis of longer periods of grain work one can observe that during grinding with grinding wheels the number of contacts with variable energy yields larger total work of grinding.

Fig. 5. Variability of micro-machining work with grains in the grinding zone for a defined grinding wheel section a: of continuous active flank, b: of discontinuous active flank

While summing up the results presented above one can conclude that for the same grinding parameters, grinding with a grinding wheel with discontinuous active flank macrogeometry is more advantageous as regards energy.

The influence of the grinding speed value \(V_s\) and longitudinal feed of the object \(V_f\) upon the variations of grinding work \(E_s\) with respect to the mean, minimum and maximum values in individual sections of the grinding zone, are all presented in Tables 1 and 2. The mean values of grinding work rise alongside with increasing speed \(V_f\), both for a grinding wheel with a continuous and discontinuous active flank. The load of grains in the grinding zone rises, as well, and grinding is characterized by high levels of energy consumption. It is also of crucial importance that at the same values of \(V_s\) and \(V_f\), the mean values of grinding work for a grinding wheel with a continuous active flank are slightly higher than the mean values of grinding work for grinding wheels with a discontinuous active flank. At the same time, the maximum grinding work values are much higher at the application of grinding wheels with a discontinuous active flank. This is depicted by the nature of the grains work, their loads at the moment of contact with the object ground, where for grains in grinding wheels with a discontinuous active flank these loads are substantially larger with bigger sections of layers being taken off, hence larger maximum grinding work values. Such a situation may result in a worsened quality as regards physical and chemical properties of the surface ground.
GRINDING ENERGY COMPARISON OF GRINDING WHEELS WITH DIFFERENT MACRO AND MICRO TOPOGRAPHIES OF SURFACE

The results of simulative examinations conducted make it possible to compare the nature of work of grinding wheels with different types of abrasive grains, which in turn have various geometrical parameters (i.e., regular boron nitride and aloxite). The grains of regular boron nitride with small radiiuses and rounded vertices possess more favorable micro-geometrical properties which facilitate the creation of a chip in the grinding zone. This is documented by the diagrams presented in Fig. 6, which point to larger sections of the machined layers and to a smaller tendency of the creation of fishes (larger Esw and smaller Esop). Hence, the proper grinding work es with grinding wheels with grains from regular boron nitride is smaller, too. Fig. 6 presents a sample comparative list of distributions of grinding work values, its components and of proper grinding work for grinding wheels with varied active flank topographies: for an aloxite grinding wheel with a regular and random distribution of grains a) and b), a regular boron nitride grinding wheel c), and an aloxite grinding wheel with a discontinuous active flank. The diagrams enable one to assess the working conditions of grains in the grinding zone. In the case of the aloxite grinding wheel with a random distribution of grains on its active flank, the local variability of work performed with individual grains is of a slightly different nature as compared with a grinding wheel with regularly distributed grains, for which proper work in the output zone is smaller, and work in the input zone is larger.

Fig. 6. Comparison of distributions of grinding work Esw Esop and of proper work es in the zone for a grinding wheel with a continuous flank a) a grinding wheel with a regular distribution of grains on the active flank; b) a grinding wheel with aloxite grains with randomly distributed grains; c) regular boron nitride with randomly distributed grains; d) aloxite with a discontinuous flank

FINAL CONCLUSIONS

The research results presented and their analysis enable one to assess the influence of the macro- and micro-topography of the grinding wheel active flank upon energetic properties of the grinding process, as well as upon the quality of the surface layer.

A regular distribution of grains facilitates a smaller diversity (variation) of the sections of layers machined with individual grains, which is the reason for a certain growth of the proper machining energy, which in turn is one of the factors deteriorating the physical and chemical properties of the surface layer.

Macro-geometry of the grinding wheel active flank, determined with its dis-continuity, has a vital influence upon the nature of the load of abrasive grains in the machining zone. In the case of grinding wheels with a discontinuous active flank, grains machine larger sections of the material while their number is decreased due to this discontinuity, owing to which the work conditions of grains change, i.e. their energy decreases; so does the path of the grain work connected with deformations of material ridging for the initiation of the chip creation process, which facilitates a decrease of the proper grinding energy.

In generation grinding of planes, a discontinuity of the grinding wheel active flank in the form of grooves has an influence upon the roughness of the surface machined. Circumferential grooves, depending of the transverse feed,
impair the topography of the surface machined and cause the occurrence of strips with a higher degree of roughness, or even the occurrence of non-machined strips. Skew grooves facilitate a creation of local protrusions on the object with a higher level of roughness, while transverse grooves constitute of form of discontinuity, which does not result in increased levels of local roughness on the object machined. A pictorial diagram of the influence of discontinuity in the form of transverse grooves on the roughness of the surface machined in the function of the share of groove in the total surface is presented in Fig. 8.

Fig. 7. Influence of grinding wheel surface discontinuity on the roughness of the surface machined, where: RaC- roughness of the surface machined with a grinding wheel with a continuous flank, RaNC- roughness of the surface machined with a grinding wheel with a discontinuous flank

REFERENCES

Królkowski T., Bałasz B., Kacalak W.: The Influence of Micro-And Macrotopography of the Active Grinding Surface on the Energy Consumption in the Grinding Process. 15TH EUROPEAN SIMULATION MULTICONFERENCE, Prague, Czech Republic 2001

BIOGRAPHY

TOMASZ KRÓLIKOWSKI was born in Szczecinek, Poland and went to the Technical University of Szczecin, where he studied fine mechanics and information theory and obtained his degrees in 1996. He is a assistant at Department of Fine Mechanics at Technical University Of Koszalin. He is interested in artificial symulations, neural networks and optimasation theory.
COMPLEX SYSTEMS SIMULATION
AUGMENTED REALITY FOR FACTORY MONITORING

Alessandro Genco
Dipartimento di Ingegneria Informatica
Università degli Studi di Palermo
Viale delle Scienze, edificio 6, 90128 Palermo, Italy
E-mail: genco@unipa.it

KEYWORDS
Ubiquitous computing, Augmented reality, Mixed reality, Context aware services, Localisation, Positioning, Factory monitoring

ABSTRACT
This paper deals with augmented reality context aware services for operators to interact with a factory environment. A model is discussed which assumes any object in reality can be managed by a virtual entity. This is based on the composition of physical resources and additional semantics to be shown when people want to interact with it. When interacting, entity and user are placed on a common layer and use a selected ontology and a peer to peer protocol. The paper initially discusses the highlights of such a model and then proposes an application for factory environments, involving operators, positioning systems, remote displays and lightening systems. On the base of operator’s data and site layout knowledge, the paper discusses how a system can drive different context aware interactions and service provisions. The paper also shows how the proposed model can be used to perform augmented reality process simulation.

INTRODUCTION
Augmented reality (AR) and mixed reality (MR) are recent research topics which extend artificial intelligence applications to the whole reality where we live (Weiser 1991) (Romero and Correia 2003). The most attractive feature of AR and MR is in that we will be enabled to interact with a surrounding reality, not only virtual, but actually a mix of real and virtual, physical and logical. An object in augmented reality is not only what an original definition can say of it; an object can become what other entities need to perceive of it, including semantic contents and actions not naturally belonging to that object (Genco 2004). Even if we perceive a surrounding environment as something natural, we often ignore that most of things we daily have to deal with, are likely to be managed as parts of some artificial entity performing its logic in some hidden system (Want et al. 2002). AR technology is often used in applications dealing with visitor guided tours in heritage sites. In (Hall et al. 2001), for instance, the authors discuss how a virtual archaeologist can explore a museum along with virtual history outdoors and hybrid physical-digital artefacts. In (Genco et al. 2004) we deal with context aware services in a castle, and how that castle could be augmented with semantics and methods for interaction with visitors. As far as AR and MR technology is concerned, many projects (Duri et al. 2001) (Long et al. 1996) are mainly based on some enabling positioning systems, such as the ones based on bluetooth (Case and Thapa 2003) (Gonzáles and García-Reinoso 2002), ultrasound, and infrared for proximity detection (Hightower and Borriello 2003). We used WiFi (IEEE 802.11x) (Lansford et al. 2001), Bluetooth and optical technologies for our experiments. This three technology combination allowed us to arrange a location aware AR system, which can work both in outdoor and indoor places. On the base of the experiences made in the field of cultural goods, we now propose a solution for industrial applications which can take advantage from the use of AR compliant applications.

THE AUGMENTED REALITY EXHIBITION MODEL
Our solution looks at AR as an environment where virtual entities cooperate. The main advantages of cooperating entities is in that they can share resources or can arrange complex services and concurrent actions. As a simple example of that we can consider a robot which is only made of an execution environment on a mobile base. The robot is not provided with any onboard video camera, but we can use the robot to explore a landscape if some camera is around. A blind robot and a video camera can cooperate to arrange a viewing robot (Figure 1). In natural life we can find the same type of cooperation when a blind man cooperate with a dog. Cooperation between robot and camera can be performed by the use of some communication technology; cooperation between man and dog is something natural which does not need any digital technology; some problems arise when natural entities need to cooperate with digital entities. Both parts need to be described according to a common semantic abstraction and use a peer to peer interaction protocol. The aim of our model is to allow entities of different type to cooperate on a peer to peer basis. Our solution is to arrange virtual entities which have the role of wrapping natural and digital resources in semantic structures. Robots, cameras, humans, dogs, and so on, become parts of augmented entities which are capable of interacting each with another according to a suitable ontology. Anything wrapped by a virtual entity
behaviours when other entities start interactions with it. Our model is called HAREM - Hybrid Augmented Reality Exhibition Model. HAREM relies on some cooperation mechanisms and a multi-layer structure to be used for the description of any kind of virtual entity in AR and MR (Genco 2004).

According to the HAREM model, entities cooperate on a semantic layer and use physical resources in a service providing fashion. The entity description structure is based on a three layer stack (Figure 2), each layer hosting a different virtual entity projection:

- a semantic projection for semantic interaction, knowledge maintenance and knowledge management,
- a middleware projection allowing entities to be implemented according to some development platform,
- a physical projection for physical interaction and physical resource management.

Semantic and middleware projections are the non-visible part of an AR entity. Physical projection accounts for physical resource management in the AR in a multithread environment, thus letting entity multi-projections interact with many other entities at a time.

The entity roles in the semantic projection are conceived to fit both real and virtual worlds as they can be seen by human eyes. A lot of behaviours can be supposed to feature beings and things when they are turned in active entities. These mostly deal with the application they are involved with. HAREM focuses on basic behaviours, the ones capable of representing the highest abstraction of the various actions for a specific applications.

The semantic layer is a structure which all logical behaviours and mechanisms should be described in. Besides maintenance and management aspects, the semantic layer describes how a virtual entity behaves when interacting with another virtual or real entity. Figure 3 show the most representative blocks in the semantic layer structure. Actually, interaction gates are managed by the middleware layer, but they are included for better understanding of the semantic layer. The highest three blocks deal with entity maintenance and fix what a given entity is, what is the meaning of
its being in AR. A change in its mission must comply with its identity as well as a given task is not allowed to be inconsistent with its mission.

When required by application ends, verification of such dependencies can be performed by means of clause verification processes.

A Harem compliant application should implement the lowest three blocks at least, namely instinct, security, and ability. Block activation can be scheduled externally by a specific selection logic, or a block can be called by an adjacent block according to a top-down or bottom-up dependency. For instance, when an interaction is requested by another entity, a communication thread is triggered by a communication gate. This thread first tries to use a default ontology for default reaction, then investigate about security risks, and finally activates its ability layer to provide the requesting entity with a service of its. A vocation block is needed and activated when an entity is capable of performing many interactions at a time or the entity was created to perform its own specific mission. A vocation block is mainly conceived for action planning and scheduling. A proactive entity could use its vocation block to exhibit its presence in an AR scenario.
The middleware layer is devoted to arrange a software solution for interactions by means of some development platform. As described below by implementation paragraphs, we used the JADE platform for our experiments. Interactions can be performed by peer to peer protocols to let different entities cooperate, or they can be performed by service provision protocols when interactions are devoted to let the physical layer provide the semantic layer with physical interaction services.

The physical layer hosts all mechanisms for physical interaction. Actuators, sensors, multimedia devices are managed in this layer according to a two step cyclic logic (Figure 4). The two steps are: exhibition and perception. The exhibition step is devoted to let an entity show its availability to the AR environment according to the role an entity wants to play. The entity may be available for service provision or it may want to request services from other entities, or it simply want to show itself in a scenario. Exhibition and perception are repeatedly performed to let an entity improve its interaction effectiveness, starting by a generic default exhibition to be followed by more specific ones. Service providing entities typically interact with many requesting entities at a time; therefore many physical interaction threads are needed. Complex systems are unlikely to be implemented according to their final configuration in the beginning.

An expansion mechanism is therefore needed to let a system grow on the base of its current resources. In the AR case, system growth is in terms of number of entities. This can be due to management improvement reasons or application requirements. In any cases Harem entity communities grow when any lack, in terms of knowledge or methods, is presented to the system. The graph in Figure 5 shows the case of an entity (C_{1411}) which needs a missing resource (R_i). First it asks for help from its default friend entity which in turn also lacks that resource. Then C_{1411} tries asking its creator C_{141}. The resource lookup process continues until an entity (C_i) is called which owns R_i. There are some options C_i can adopt in order to provide C_{1411} with R_i. One of these is to create a new entity (C_{15}) whose mission is to manage R_i on entity request. According to this creation mechanism, each entity in an AR system can play both roles of creator and creature. Triangles in the graph stand for entities which participate in the search as creators.

**Harem for Simulation**

Human beings and physical things are described in augmented reality as virtual entities according to a common semantic structure. A human behaviour is described as a virtual entity behaviour. A physical thing behaviour is also described as a virtual entity...
behaviour. This approach has two main advantages: natural and virtual beings can interact between them as they were belonging to the same entity kind; when dealing with simulation applications, each entity component can be instantiated by a real, physical, or natural part as well as by a software made virtual object.

Their semantic structures are similar by definition, because the behaviour of a wrapped object must be known to the entities that object will interact with. Interaction is always performed on the base of a known ontology and a protocol which is a peer to peer one in the case of interaction between entities, and a service provision protocol when entities interact with real physical objects. It turns out to be very simple to replace a real object with a functionally equivalent virtual one. As it will be emphasized in the case study paragraphs, operators, processes, and whatever other real element can be replaced by a functionally equivalent virtual object. Obviously simulation can be performed as for the only behaviours known to the system.

**FIPA Agent Based Implementation**

Semantic projection is being implemented according to the intelligent agent paradigm as stated by the FIPA standard (FIPA SC00001L). A FIPA agent can pursue a goal by playing several roles, each dealing with a particular sub-goal. HAREM entities are implemented as FIPA agents with application specific operating roles. Here we deal with five roles:

- **receiver**: a role, which accepts entity calls for cooperation;
- **vocation**: a role, which selects incoming calls and schedules other roles to be activated;
- **ability**: a role, which looks for external resources for task accomplishment;
- **survival**: a role, which is devoted to safety reactions against suspected calls. It also implements fault tolerance policies;
- **instinct**: a role for first exhibition and default reactions to generic calls.

Diagram in Figure 6 describes the HAREM role interaction protocol according to the representation proposed by Odell (Odell et al. 2001). Each call for cooperation to a HAREM entity is first processed by receiver, then forwarded to survival and vocation. If survival detects some danger, it alerts vocation. If no danger is detected, vocation selects the application specific operating roles to comply with the request. Communication between vocation and the other roles is performed according to the FIPA specifications on Agent Interaction Protocols (FIPA XC000029F) and to the communicative act semantics in (FIPA SC00037F). Each role can send a request to ability for external resource lookup. This is accomplished by a request sent by ability to other AR entities. Finally, vocation sends the request to the instinct default role only if no other operating role can comply with it.

HAREM virtual entities are coded in Java on the JADE platform (Bellifemine 2003). HAREM is therefore FIPA compliant.

**THE FACTORY CASE STUDY**

We are currently developing a HAREM based system for applications in factory environments. This project has many similarities with another project (Genco et al. 2004) dealing with AR in heritage sites. This project aims at turning a factory building into augmented reality for interaction between operators and a surrounding factory environment. To this end, operators need to be provided with some mobile device, as for instance, a PDA or a cellular phone and the Bluetooth connectivity feature.

**Factory Context Aware Services**

Our application system uses AR featuring interactions to provide a selected arrangement of context aware services. An industrial place involves a lot of interactions between operators and the surrounding environment. Here we discuss some of these, which are just representative of a wider set of interactions.

Figure 6: HAREM Agent Interaction Protocol
potentially manageable by means of AR technologies. Let us consider the following list of context aware services:

- **Operator positioning**: the position of each operator is traced and stored, along with his personal data.
- **Lightening system activation by operator presence**: lights in a room is switched on when an operator is in.
- **Real time monitoring of production processes**: a mobile or wall display can be used by the system to show real time video representations of a process, or any related technical information, to a neighbouring operator.
- **Message exchange between operators**: operators can exchange messages with one another, receive instructions from any industrial process knowledge base, and orders from superiors.
- **Guided path services**: an operator can be directed by his mobile device to a place where to perform his next task.

All of these provide some effective facilities to the daily activity of a factory or industrial place. What the system need is some suitably arranged logic and some devices which are typical of AR. As for logic, we take advantage from our HAREM model as discussed before. The device collection includes some mobile wireless devices (Bluetooth, and WiFi), some digital cameras, some wall displays, and at least one computer to manage all context aware service.

**Operator Positioning**

Operator positioning service makes use of a Site Positioning System (SPS), which performs position tracing of each operator and stores 2 or 3 fixed coordinates of factory layout and relevant objects. SPS is based on Bluetooth and image processing techniques.

**Bluetooth Positioning**

We suppose each operator has been provided with some Bluetooth mobile station (BT-MS). Some Bluetooth base stations (BT-BS) are placed in the surrounding area. Positioning is made possible by the use of the Host Controller Interface (HCI) layer of Bluetooth protocol (Case 2003), namely: `Get Link Quality` returns a quality measure of the link between a BT-BS and a BT-MS, as an integer in the range 0-255, and `Read RSSI`, returns a signal intensity measure.

We used Bluetooth devices of power class two, thus covering a range of ~10 m. The worst case accuracy was ~10 m. Positioning was performed in two steps: measurement and hyperbolic triangulation.

Our trials were carried out at our Department of Computer Engineering in Palermo. We put Bluetooth devices (cellular phones) in several rooms in the department. On the base of these measures, we performed hyperbolic triangulation to get an evaluation of the BT device positions. Unfortunately, class two BT device are not very reliable in evaluating distances. So we are developing a Fuzzy-neural system to allow an algorithm to use a significant sample of measures. These are first used to build the geometrical knowledge of a place, and then to compare real time measurement with the learned values.

**Optical Compass Positioning**

Proximity detection is often not enough for context aware service provision. When an operator is near an object, he can be there, either to deal with it, or he is just crossing to go elsewhere. In the first case the environment should activate the corresponding information service, in the second case it has just to ignore operator proximity. If only proximity detection is allowed, the two situations cannot be distinguished. So we provide our SPS system with a compass detection functionality for operator orientation recognition.

The compass detection functionality is implemented by image processing techniques. Each operator at site entrance is provided with a coloured marker. Compass detection is accomplished by marker image detection. According to the detected marker image, operator orientation can be recast. Some preliminary experiments on the detection of coloured regions were carried out by adopting an image segmentation technique based on dynamic threshold of colour features (Vitabile et al. 2001). To this end, the image data under analysis is mapped into the HSV (Hue-Saturation-Value) colour space. The HSV image is therefore analysed on a pixel-by-pixel basis, and only pixels falling into the appropriate range of hue and saturation are retained. This technique needs to be refined in order to implement the detection of other important features, like proximity estimation. To this aim, multi-zone coloured regions will be taken into consideration, with a colour code associated to proximity values. This approach can be used for operator positioning refinement. Namely, a marker can be designed with multi-zone coloured regions, so that a proximity estimation can be drawn by processing the detected marker image and recognizing its colour pattern.

**Wi-Fi Based Multimedia Data Exchange**

Some services require multimedia data transmission. For example, real time process monitoring may require information movie displaying, as well as process simulation.

Bluetooth technology is not suitable for multimedia data exchange, so we use the Wi-Fi (IEEE802.11x) technology. Its compatibility with Bluetooth technology has been investigated (Lansford et al. 2001).

**The SPS Entity**

The SPS entity has an operating role (Figure 7) for operator Bluetooth positioning (BT_Role) and an operating role for operator compass detection (C_Role). When an operator approaches some AR equipped object, the SPS Bluetooth role detects proximity, and the Compass role detects operator orientation. According to the results of these

78
detections, an SPS request triggers an overall interaction for context aware service providing.

![Figure 7. SPS Role Interaction Protocol](image)

Both Bluetooth and Compass roles exploit external resources (Bluetooth hardware, digital camera, etc.) by means of SPS entity ability invocation. More precisely, when SPS BT_Role activates BT_Positioning(), it needs to use some physical BT-BS available in the environment. In our AR environment, a Bluetooth entity exists with methods for physical BT-BS management. BT_Positioning() does not include any reference to the Bluetooth entity, so it sends a request to the SPS ability role. This has a reference to the Bluetooth entity, and sends a call for cooperation to it. Optical devices and image processing for compass detection are managed the same way.

A UML representation of interaction among the SPS entity roles is presented in Figure 7. The SPS vocation sends a request to the BT_Role for Bluetooth positioning. BT_Role performs the requested task and answers with an inform act on position data. Next the SPS vocation sends a request to C_Role for compass detection and positioning refinement. The C_Role performs the requested operation and informs vocation about the results. Vocation processes these results and sends a request to other entities in the environment for context dependent interaction triggering. Ability invocations for external resource management are also showed.

**Automatic Lightening**

A set of spotlights is placed in a room and near the most relevant objects. Spotlights must be automatically switched on when any operator is around. This interaction case involves a Room entity, an SPS entity, a Light entity. When the SPS entity detects some operator in the room, it sends a message to the Room entity. The message carries data about operator position and orientation. The message is first taken by the Room receiver role, and then sent to survival and vocation. If survival does not detect any danger, vocation processes SPS request. Based on position and orientation information in the message, Room entity vocation decides whether lights have to be switched on or not. If lights must be switched on, the Room entity sends a request to the Light entity for lightning system activation. Light entity receives the request message the same way as described before. If Light entity survival does not detect any danger, its vocation activates LightOn(), which switches on the spotlights. LightOn() activation requires physical spotlight management, which is accomplished by the SPS entity ability invocation.

**Real Time Monitoring of Industrial Processes**

This service may need some multimedia facilities to allow an operator to monitor a production or a control process, and compare it with a virtual optimal performance reconstruction. Hardware for this case can be Wi-Fi technology, digital camera, remote display, mobile terminal, along with what needed for operator positioning. Similarly to the previous cases, all is managed by entities and their operating roles. Here we want the AR environment to display multimedia information on a wall display or operator’s device when the right operator enters the place where a process is running. The same service can be requested by an operator as a process call. Moreover, we want this information to be customized, according to the operator’s role and authorisation level. For example, we want some details to be hidden to generic operators. Entities involved in this case are: the Room entity, the SPS entity and the Wi-Fi entity (Figure 8), and other entities dealing with multimedia devices. When an authorised operator is detected in the room by the SPS system, SPS entity sends a request to the Room entity. Identification data about the detected operator are enclosed in the request message. Based on these data, the Room entity selects multimedia information (MMI) to be displayed, and sends a request to the Wi-Fi entity to trigger chosen MMI display.

**System Test by Real Part Simulation**

![Figure 8. Co-operation Diagram for Context Triggered Information Display](image)
Once fixed the application environment and the operator behaviours to be known to the system, some services of the pervasive system for factory monitoring were tested in our laboratories at the department of computer engineering in Palermo. Our trials were carried out on a simulation basis. We used the layout description of our department as it was the one of a factory and we played the role of factory operators. Early simulations were conducted by replacing operators with virtual entities which were arranged to show the same behaviours an operator can show in a factory. In practice, the most relevant operator behaviour is its walking inside a factory building. Therefore, the hardware parts of the SPS entity were replaced by path generator virtual objects. On the basis of the data generated by these objects and operator personal data the system was able to simulate spotlight switching and remote display output.

CONCLUSIONS

We presented an AR application to factory environment, which takes advantage by the use of our AR entity description model. This assumes that any real object can be augmented if wrapped by a virtual entity which implements active behaviours. Three main instances have been discussed which were designed according to the same cooperation model between virtual AR entities and their operational roles. Namely, an operator positioning service, an automatic lightening service, and a service for real time monitoring of industrial processes have been discussed. Other factory context aware services, such as operator message exchange, and guided path, were only listed because of paper size restrictions. However, we want to emphasize the evidence that any kind of context aware service in AR can be designed according to the same entity description model. We are currently using the same approach for AR applications also in the field of cultural goods.

ACKNOWLEDGEMENTS

This work has been partly supported by the CNR (National Research Council of Italy) and partly by the MIUR (Ministry of the Instruction, University and Research of Italy).

BIOGRAPHY

Alessandro Genco was born in Cagliari, Italy, and went to the University of Palermo, where he studied mathematics and obtained his laurea degree in 1972. He is an associate professor with the Department of Computer Engineering in Palermo, where he leads the PHD school of Computer Engineering. He is also with the Faculty of Motion Sciences where he covers the position of delegate for technological services. His main research interest is in distributed systems. Recently, he started dealing with ubiquitous computing, especially in the field of augmented reality and context aware services.

REFERENCES

Genco A., HAREM: the Hybrid Augmented Reality Exhibition Model, WSEAS Transactions on Computer, Issue 1, Volume 3, January 2004, ISSN 1109-2750
Genco A. et al. Context Aware Services for Tourists in a Late Middle Age Castle, to appear in the proceedings of the 8th World Multi-Conference on Systemics, Cybernetics and Informatics July 18-21 2004, Orlando, Florida, USA
NEURAL NETWORKS BASED TOOLS FOR PRECEDENCE CRITERIA DEFINITION IN INFRASTRUCTURE MAINTENANCE MANAGEMENT

Abhijit Deshmukh
Dept. Mechanical & Industrial Eng.
University of Massachusetts
Amherst, MA 01003 USA
deshmukh@ecs.umass.edu

Lorenzo Fedele
University of Rome ‘La Sapienza’
Dept. Mechanics & Aeronautics
00184 Via Eudossiana 18, Rome Italy
lorenzo.fedele@uniroma1.it

Giuseppe Fiacco
C.N.I.M.
Via Barberini 68
00187 Rome, Italy
g.fiacco@cnim.it

ABSTRACT

Evaluation subjectivity errors represent a serious problem for all those procedures based on human visual inspection in the field of maintenance activities. In this sense, artificial neural networks can successfully be used as an effective and robust decision making instrument to support inspectors in defect judgement. This work is focused on human judgement simulation for railways infrastructures defects, particularly bridges. The estimated defect parameter is the evolution factor which is representative of the urgency of intervention on the considered defect, in terms of involved resources and time delay. Moreover it is demonstrated the effective applicability of the neural network approach for maintenance management activities. The results show that neural networks, trained with experimentally obtained sample cases, have a strong potential for decision making supporting in several maintenance management application fields.

AN EXAMPLE OF MULTI-PATTERN INFRASTRUCTURE SYSTEM MAINTENANCE MANAGEMENT: THE DOMUS PROJECT.

Most of maintenance management enterprises, especially those focused on transportation structures, are based on computer aided management systems, which support decision making activities. All maintenance activities can be summed up as follows:
- diagnosis;
- inspection;
- structure data collection;
- structure data analysis;
- repair or replacement;
- reinforcement.

Each activity, except for the last two in the above list, is batched into standard procedures for structures evaluation (SEP), that analyze available data to find out existing degradation processes or to predict new ones. They are successfully used for technical analysis of different material properties, repair procedures and monitoring systems. The bridge SEP system implemented into the Italian Railways Maintenance Society, RFI-Rete Ferroviaria Italiana S.p.A., is the DOMUS Project, developed by CNIM-Comitato Nazionale Italiano per la Manutenzione.

Its name is an acronym concerning Standard Structure Diagnosis and Maintenance Management and consists of the implemented instruments listed below:
- a complete structures data-base, which contains all construction, environmental, seismic and maintenance data for each structure;
- standard schedules to batch structures into homogeneous groups;
- a detailed defect catalogue which is supported by photos and schemes;
- a visual inspection procedure supported by tablet-pc;
- an empirical priority ranking evaluation algorithm to assign a numerical index value to each structure which is representative of its defectiveness level.

The empirical index value is the result of the following formula:

\[
I = \frac{\sum_{j=1}^{N_{ce}} V_{Defj}}{\sum_{j=1}^{N_{ce}} V_{Dref}}
\]

where \( V_{Defj} \) represents the effective deficiency of the j-component defined as follows:

\[
V_{Defj} = K_{ij} \cdot \sum_{j=1}^{nd} B_{ij} \cdot K_{2el} \cdot K_{3el} \cdot K_{4el}
\]

\( nd \) = number of observed defect types
\( K_{ij} \) = j-component importance factor for the structural system
\( N_{ce} \) = total number of components in the considered structure
\( B_{ij} \) = failure importance of the defect i, which expresses its potential effects on the observed structure safety and durability
\( K_{2el} \) = maximum value of the intensity factor indicating the degree or intensity of the defect type i, among those observed on all j-type components
\( K_{3el} \) = maximum value of the extension factor concerning extension and expected propagation of the defect type i, among those observed on all j-type components
\( K_{4el} \) = maximum value of the evolution factor that is representative of the urgency of intervention on the defect type i, among those observed on all j-type components

\( V_{Dref} \) = maximum value for the effective deficiency of all components
Concerning Subjectivity Error In DOMUS Implemented Visual Inspections

As described in the previous chapter, the Domus priority ranking index is based on the numerical value of three adimensional parameters, $K_2$ or intensity, $K_3$ or extension and $K_4$ or evolution factor, related to each observed defect on the considered structure. These values are assigned by Bridge Administrator's inspectors according to the correspondence to the defect catalogue scheduled descriptions and are based on logical associations due to their personal attitudes and skills. The main problem that affects this inspection procedure and, most generally, every visual inspection, is the inspector’s subjectivity error. The main causes are listed below:

- technical background;
- specific accumulated experience;
- problem detection sensitivity;
- attention level;
- job motivation.

Consequently, subjectivity errors involve a lost of useful information, heterogeneous judgements, decisions that are not complying with actual structure requirements, unacceptable risks for structure safety and integrity, and economic losses. It would be necessary to design an efficient instrument to support decision making for such parameters. It should be able to simulate human logical associations regardless of particular crisp or empirically approximated relations concerning involved factors. The chosen instrument, fairly implemented into the engineered Domus system, are the artificial neural systems. The main neural networks features, that make them particularly eligible for inspector’s judgement simulation, are:

- experimentally demonstrated capability of learning by examples;
- real time response capability;
- brain cell-like approach in information processing;
- eligibility to classify defects parameters according to the fuzzy logic approach, as demonstrated by the lack of crisp border values in the scheduled descriptions;
- noise reduction capability.

$K_4$ Parameter As Object Of The Neural Networks Simulation

Regarding both economical and strategic reasons, the most important adimensional parameter, among those which completely define the actual defect state, is the $K_4$ one. It is due to several reasons as shown below.

First, $K_4$ values, despite of $K_2$ and $K_3$ ones, are assigned according to the same criteria for all defect types, so that $K_4$ available information level is lower than other parameters one. Moreover, $K_4$ values may directly affect the entire structure maintenance planning, which is based on structure or, at least, structure element, not single defect concerning. Therefore, while partial or total traffic restrictions are planned for certain defects critical $K_4$ values, it may happen that a heavy overrated $K_4$ assessment would involve unacceptable economic losses for both structure Administrator and Users.

On the other hand, every underrating of the considered parameter would involve weighty problems in terms of safety and structure integrity risks.

It eventually must be noted that $K_4$ even depends on defect intensity and extension states, so that it partially depends on $K_2$ and $K_3$ parameters.

Neural Network Design Features

In order to make neural network systems base their output values on the same inspector’s logical associations, all factors that influence his evaluation are included into the neural network defect input vector.

They are related to the considered structure, structure element, element component and component defect, as listed below:

1.) Structure inputs
   - structure family, that includes the main bridge construction typologies;
   - vertical elements number, including piers and abutments;
   - surface waterproofing treatment presence, which is particularly significant for humidity related defects;
   - meteoric water drainage system presence, which is significant for humidity involved defects;
   - seismic category, which is representative of seismic activity magnitude in the area where the considered structure is located;

2.) Element inputs
   - element type, that corresponds to vertical or horizontal geometry;
   - element subtype, that expresses the considered element construction type;
   - material type, that defines the element type of material;

3.) Component inputs
   - type of component, that defines the considered component typology;
   - $K_1$ parameter, which expresses the considered component weight on the structural system;

4.) Defect inputs
   - defect type, that defines the typology of the considered defect;
   - $B$ coefficient, which is significant of the considered defect effects on structure safety and durability;
   - $K_2$ parameter;
   - $K_3$ parameter.

It is necessary to note that, such factors like bridge traffic level and environmental setting are not included into the input vector because of the lack of those data inside of the structure data-base when the processed defects have been overseen, preventing inspectors to be influenced by them.

The total number of available data for this work amounts to 467, corresponding to as much overseen defects on 17 railways bridges.

Data allocation between training and validation sets is based on defect output values as shown in the figure below.
<table>
<thead>
<tr>
<th>$K_a$ value</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>training</td>
<td>135</td>
<td>118</td>
<td>114</td>
<td>367</td>
</tr>
<tr>
<td>validation</td>
<td>37</td>
<td>32</td>
<td>31</td>
<td>100</td>
</tr>
<tr>
<td>total</td>
<td>172</td>
<td>150</td>
<td>145</td>
<td>467</td>
</tr>
</tbody>
</table>

Figure 1: experimental data distribution

As resulting from preliminary tests series, certain neural network parameters have been chosen and fixed during the entire following experimentation.

Feed-forward multilayer perceptron architectures have been chosen with back-propagation training algorithm, because of their flexibility and capability to simulate almost every kind of non-linear relation.

According to the actual state of the art, all chosen neural network parameters are summed up in the list below.
- hidden layer number [1; 3];
- neurons per layer [3; 15];
- transfer functions: sigmoid, bipolar sigmoid;
- weight training function: Levenberg-Marquardt;
- error function: Sum of Square Errors;
- number of training epochs: 1000;
- number of produced networks for each architecture: 5.

All architectures have been changed according to their number of layers, number of neurons per layer and type of transfer function, modifying one parameter per test only. Five comparison parameters have been selected to check architectures results.

1.) $\text{SSE}_{\text{train}}$, which represents the Sum of Square Errors value resulting at the end of the training session and is significant for network capability to memorize, in order to produce output values much close to the corresponding target values.

2.) $\text{SSE}_{\text{valid}}$, which is the Sum of Square Errors regarding validation data and is significant for network capability to produce good output values without any information about the corresponding target values.

3.) $\%E<0.1$, or percentage value of validation errors lower than 0.1, that is important to evaluate networks precision.

4.) $\%E<0.01$, or percentage value of validation errors lower than 0.01, that is important to evaluate networks capability to produce discrete output values.

5.) $\% \text{ correct}$, validation correct output values percentage, which is a measure of networks capability to evaluate the considered parameter $K_a$.

Validation output values have been considered correct while confirming one of the following equations:

$\text{Out}_{\text{rn}} \cdot a = \text{target}$

$\text{Out}_{\text{rn}} \cdot a = \text{target} + 1$

where $\text{Out}_{\text{rn}} \cdot a$ is the closest integer approximated value for each network output. While the first equation corresponds to the exact target value, the second one represents a different value due to experimental data subjectivity errors, which overrates target values in favour of safety and structure durability even if it is limited in order to avoid economic planning problems.

**EXPERIMENTAL TESTS**

While neural networks constitute a black box instrument, it is almost impossible to make an effective and rigorous control on them. That is confirmed, for example, by NN operator’s impossibility to monitor weight values and put them in relation to the corresponding output values. In order to rationalize this experimentation, one parameter has been changed among neurons per layer, number of layer and type of transfer function, for each test, basing successive choices on the previous test results.

**TEST #1**

During the first test four single hidden layer architectures have been compared with sigmoid transfer function and a range of neurons per layer from 3 to 15 as shown below.

![Architecture features schedule for TEST 1](image)

In the figures below all results are shown.

![Architecture results for TEST 1](image)

![Architecture results for TEST 1](image)
Fig 3 shows very good values for the percentage of correct outputs which are, therefore, associated to elevated SSE values both for training and validation. Moreover, low \( %E<0.01 \) values put in evidence a short capability to produce discrete outputs. Dealing with Figure 4, the overfitting phenomenon is evident, especially for 1D architecture; it is due to excessive number of neurons per layer and make networks improve their capability to memorize only. That bad behaviour is confirmed by 1D Output unacceptable values. A particular attention merits NN 3 for 1C architecture, which has produced good values for all parameters except validation SSE. It is due to certain negative and overrating outputs. According to the considered output results, 15 neurons layers have been excluded by next tests.

**TEST #2**

In this test sigmoid function has been substituted with the bipolar sigmoid one, with the same architectures except for the last one.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Number of neurons</th>
<th>Transfer function.</th>
<th>Hidden layers</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 A</td>
<td>3</td>
<td>Bipolar sigmoid</td>
<td>1</td>
</tr>
<tr>
<td>2 B</td>
<td>5</td>
<td>Bipolar sigmoid</td>
<td>1</td>
</tr>
<tr>
<td>2 C</td>
<td>10</td>
<td>Bipolar sigmoid</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 5: architectures features schedule for TEST 2

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>( \text{SSE}_{\text{train}} )</th>
<th>( \text{SSE}_{\text{valid}} )</th>
<th>% ( E&lt;0.1 )</th>
<th>% ( E&lt;0.01 )</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 A</td>
<td>1</td>
<td>24</td>
<td>21.08</td>
<td>78</td>
<td>44</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>44.77</td>
<td>30.78</td>
<td>26</td>
<td>2</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8.46</td>
<td>33.24</td>
<td>64</td>
<td>17</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>21.21</td>
<td>47.77</td>
<td>39</td>
<td>18</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4.02</td>
<td>33.58</td>
<td>70</td>
<td>45</td>
<td>93</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>( \text{SSE}_{\text{train}} )</th>
<th>( \text{SSE}_{\text{valid}} )</th>
<th>% ( E&lt;0.1 )</th>
<th>% ( E&lt;0.01 )</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 B</td>
<td>1</td>
<td>17.77</td>
<td>37.07</td>
<td>61</td>
<td>14</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>24</td>
<td>21.84</td>
<td>33</td>
<td>6</td>
<td>95</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>35.21</td>
<td>50.32</td>
<td>18</td>
<td>0</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2.1</td>
<td>34.36</td>
<td>73</td>
<td>73</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>34.32</td>
<td>50.98</td>
<td>47</td>
<td>4</td>
<td>87</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>( \text{SSE}_{\text{train}} )</th>
<th>( \text{SSE}_{\text{valid}} )</th>
<th>% ( E&lt;0.1 )</th>
<th>% ( E&lt;0.01 )</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 C</td>
<td>1</td>
<td>23.68</td>
<td>45.77</td>
<td>55</td>
<td>31</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.12</td>
<td>30.72</td>
<td>48</td>
<td>38</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2.1</td>
<td>91.84</td>
<td>26</td>
<td>24</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>32.47</td>
<td>33.71</td>
<td>54</td>
<td>31</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2.1</td>
<td>61.44</td>
<td>18</td>
<td>10</td>
<td>80</td>
</tr>
</tbody>
</table>

Figure 6: 2A, 2B and 2C architectures results

Regarding the 3-neurons architecture an improvement is shown by the comparison of Fig 3 and Fig 6 results. Among 2B networks very good values have been obtained for the fourth network as shown in the graph below.

**TEST #3**

During the third test, nine architectures have been compared, all constituted by two hidden layers and bipolar sigmoid function type.

Figure 7: 2B4 graphical comparison between target and approximated outputs for validation data

By comparing 1C and 2C architectures it can be noted an increase in the second one according to the correct output percentage and the capability to generalize as pointed out by the following figure.

Figure 8: \( \text{SSE}_{\text{valid}} \) comparison for 1C and 2C architectures

According to the second TEST results, bipolar sigmoid functions have been used for all the following tests.
<table>
<thead>
<tr>
<th>Architecture</th>
<th>Number of neurons I layer</th>
<th>Number of neurons II layer</th>
<th>Transfer function</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 A</td>
<td>3</td>
<td>3</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 B</td>
<td>5</td>
<td>5</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 C</td>
<td>10</td>
<td>10</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 D</td>
<td>3</td>
<td>5</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 E</td>
<td>5</td>
<td>3</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 F</td>
<td>10</td>
<td>5</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 G</td>
<td>5</td>
<td>10</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 H</td>
<td>10</td>
<td>3</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>3 I</td>
<td>3</td>
<td>10</td>
<td>bipolar sigmoid</td>
</tr>
</tbody>
</table>

Figure 9: architecture features schedule for TEST 3

Results schedules are shown below.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 A</td>
<td>1</td>
<td>125,53</td>
<td>68</td>
<td>5</td>
<td>4</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>51,47</td>
<td>39,71</td>
<td>26</td>
<td>21</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>5,01</td>
<td>100,14</td>
<td>74</td>
<td>70</td>
<td>96</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>230</td>
<td>68,34</td>
<td>0</td>
<td>0</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>79,91</td>
<td>86,43</td>
<td>20</td>
<td>9</td>
<td>63</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 B</td>
<td>1</td>
<td>2,38</td>
<td>47,65</td>
<td>63</td>
<td>35</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>14,7</td>
<td>45,24</td>
<td>35</td>
<td>3</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>62,03</td>
<td>31,04</td>
<td>52</td>
<td>16</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2,1</td>
<td>29,56</td>
<td>62</td>
<td>50</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2,16</td>
<td>1155,8</td>
<td>66</td>
<td>59</td>
<td>93</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 C</td>
<td>1</td>
<td>2,1</td>
<td>64,86</td>
<td>42</td>
<td>42</td>
<td>74</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2,1</td>
<td>68,79</td>
<td>58</td>
<td>53</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>130,54</td>
<td>63,77</td>
<td>6</td>
<td>2</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>176,21</td>
<td>81,73</td>
<td>1</td>
<td>0</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2,1</td>
<td>53,16</td>
<td>55</td>
<td>40</td>
<td>90</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 D</td>
<td>1</td>
<td>9,82</td>
<td>76,55</td>
<td>67</td>
<td>59</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10,05</td>
<td>52,66</td>
<td>39</td>
<td>32</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2,1</td>
<td>181,69</td>
<td>66</td>
<td>64</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>215,51</td>
<td>76,83</td>
<td>2</td>
<td>0</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2,52</td>
<td>43,78</td>
<td>59</td>
<td>49</td>
<td>84</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 I</td>
<td>1</td>
<td>47,68</td>
<td>56,66</td>
<td>15</td>
<td>14</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16,28</td>
<td>20,48</td>
<td>55</td>
<td>15</td>
<td>98</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 E</td>
<td>1</td>
<td>117,34</td>
<td>102,12</td>
<td>6</td>
<td>0</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2,1</td>
<td>23,68</td>
<td>75</td>
<td>73</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>196,71</td>
<td>66,41</td>
<td>7</td>
<td>0</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>28,96</td>
<td>57,27</td>
<td>26</td>
<td>5</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>200,52</td>
<td>54,5</td>
<td>0</td>
<td>0</td>
<td>69</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 F</td>
<td>1</td>
<td>2,1</td>
<td>51,03</td>
<td>61</td>
<td>56</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2,1</td>
<td>37,22</td>
<td>61</td>
<td>57</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>5,06</td>
<td>26,08</td>
<td>49</td>
<td>11</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2,1</td>
<td>50,42</td>
<td>63</td>
<td>58</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2,1</td>
<td>38,59</td>
<td>78</td>
<td>78</td>
<td>95</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 G</td>
<td>1</td>
<td>210,03</td>
<td>77,22</td>
<td>0</td>
<td>0</td>
<td>66</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>176,68</td>
<td>56,03</td>
<td>0</td>
<td>0</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>219,1</td>
<td>74,84</td>
<td>0</td>
<td>0</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>214,68</td>
<td>66,37</td>
<td>0</td>
<td>0</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>118,88</td>
<td>73,97</td>
<td>10</td>
<td>10</td>
<td>69</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE$_{train}$</th>
<th>SSE$_{vald}$</th>
<th>% $\varepsilon$ &lt; 0.1</th>
<th>% $\varepsilon$ &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 H</td>
<td>1</td>
<td>2,1</td>
<td>106,84</td>
<td>55</td>
<td>48</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>5</td>
<td>93,46</td>
<td>53</td>
<td>19</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3,49</td>
<td>154,32</td>
<td>60</td>
<td>46</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2,1</td>
<td>45,74</td>
<td>75</td>
<td>75</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2,1</td>
<td>1484,4</td>
<td>61</td>
<td>60</td>
<td>79</td>
</tr>
</tbody>
</table>

Figure 10: TEST # 3 architectures results

By analyzing the first three architectures, which are characterized by the same number of neurons for each layer, it is evident that results regarding percentages of correct outputs are partially good, while high SSE values both for training and validation are noted, sometimes associated to overrating output values. In this sense, it can be affirmed that these kind of architectures are characterized by not good memorizing and generalizing capabilities, excluding them by the following TESTS. Dealing with 3D and 3E architectures, which are constituted by three and five neurons per layer, better results are pointed out in percentages of correct outputs, associated, therefore, to poor SSE values in training and validation and consequently have been excluded by the following TESTS. The same decision has been taken for 3F architecture, even if its second network is characterized by very good results in all parameters, with very low SSE values and high percentages for 0.1, 0.01 errors and 98% of correct outputs. Architecture 3G shows, on the contrary, good results for all considered parameters. It is confirmed by the good output approximations for the network 3G5, which are compared.
to the corresponding target values in the graph below.

![Graph showing K4 target and output](image)

Figure 11: 3G5 graphical comparison between target and approximated outputs for valid data.

It has no errors in the highest K₄ class even if there are two underratings for the second class and three illigical answers for the first class. While the last two architectures haven’t produced satisfactory results, the 3G architecture has been considered for the last TEST only.

**TEST # 4**

During the fourth TEST two architectures have been considered, both with three hidden layers and bipolar sigmoid function type.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Number of neurons I layer</th>
<th>Number of neurons II layer</th>
<th>Number of neurons III layer</th>
<th>Transfer function</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 A</td>
<td>5</td>
<td>10</td>
<td>5</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>4 B</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>bipolar sigmoid</td>
</tr>
</tbody>
</table>

Figure 12: architecture features schedule for TEST 4

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE_train</th>
<th>SSE_valid</th>
<th>% ε &lt; 0.1</th>
<th>% ε &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 A</td>
<td>1</td>
<td>127.76</td>
<td>47.67</td>
<td>12</td>
<td>12</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2,1</td>
<td>25.42</td>
<td>77</td>
<td>75</td>
<td>96</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2,1</td>
<td>31.78</td>
<td>67</td>
<td>62</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2,1</td>
<td>55.68</td>
<td>68</td>
<td>61</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2,1</td>
<td>36.54</td>
<td>68</td>
<td>57</td>
<td>86</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>SSE_train</th>
<th>SSE_valid</th>
<th>% ε &lt; 0.1</th>
<th>% ε &lt; 0.01</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 B</td>
<td>1</td>
<td>2,1</td>
<td>32.54</td>
<td>77</td>
<td>75</td>
<td>95</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2,1</td>
<td>26.72</td>
<td>78</td>
<td>76</td>
<td>96</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2,1</td>
<td>72.55</td>
<td>57</td>
<td>49</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2,1</td>
<td>20.72</td>
<td>68</td>
<td>63</td>
<td>95</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2,1</td>
<td>21.98</td>
<td>75</td>
<td>74</td>
<td>98</td>
</tr>
</tbody>
</table>

Figure 13: results schedule for 4A and 4B architectures

Dealing with the first architecture, all parameters, except for the first network, can be considered satisfactory. The first network, in fact, stopped its training after 237 epochs because of a relative minimum. Much better are 4B results while four times on five they show percentages of correct output values largely over 90%. SSE values are, in fact, very low confirming the robustness of this architecture. By comparing the last two networks SSE values with all of the best networks for each of the previous architectures, it is clear that they represent the lowest of the entire experimentation as shown below.

![SSE values comparison](image)

Figure 14: SSE lowest values comparison

The following graph shows output data for network 4B5, characterized by the best percentage of correct output values.

![4B5 outputs graphical comparison](image)

Figure 15: 4B5 outputs graphical comparison
**VALIDATION TEST**

While using trained neural networks for actual data evaluation, they may produce unacceptable results even with good output values both for training and validation sets. It is due to excessive similarity between training and validation data, that prevent their generalizing capability. In order to prevent the described phenomenon occurrence, a validation test has been carried out by reducing the number of training data and heterogeneously distributing them among $K_4$ classes as shown below.

<table>
<thead>
<tr>
<th>$K_4$ value</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>training</td>
<td>78</td>
<td>86</td>
<td>91</td>
<td>255</td>
</tr>
<tr>
<td>validation</td>
<td>94</td>
<td>64</td>
<td>54</td>
<td>212</td>
</tr>
<tr>
<td>total</td>
<td>172</td>
<td>150</td>
<td>145</td>
<td>467</td>
</tr>
</tbody>
</table>

Figure. 16 : data distribution

Two architectures have been produced with one and three layers as shown below.

<table>
<thead>
<tr>
<th>architecture</th>
<th>Number of neurons I layer</th>
<th>Number of neurons II layer</th>
<th>Number of neurons III layer</th>
<th>Transfer function</th>
</tr>
</thead>
<tbody>
<tr>
<td>VB</td>
<td>5</td>
<td>-</td>
<td>-</td>
<td>bipolar sigmoid</td>
</tr>
<tr>
<td>VD</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>bipolar sigmoid</td>
</tr>
</tbody>
</table>

Figure 17 : architectures features

The following schedule shows results for architecture VB.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>$SSE_{train}$</th>
<th>$SSE_{total}$</th>
<th>% $% &lt; 0.1$</th>
<th>% $&lt; 0.01$</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11,59</td>
<td>92,11</td>
<td>33,49</td>
<td>22,64</td>
<td>92,45</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>23,04</td>
<td>120,68</td>
<td>37,73</td>
<td>3,3</td>
<td>82,54</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>7,47</td>
<td>117,1</td>
<td>52,83</td>
<td>18,86</td>
<td>80,18</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>6,68</td>
<td>226,77</td>
<td>46,69</td>
<td>33,96</td>
<td>79,24</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>40,88</td>
<td>84,8</td>
<td>13,2</td>
<td>3,77</td>
<td>85,84</td>
<td></td>
</tr>
</tbody>
</table>

Figure 18 : results schedule for VB architecture

As described in the schedule above, one hidden layer architecture shows satisfactory values especially for correct outputs percentages like 92.45 for VBI network without relevant differences with architecture 2B, which have the same architecture parameters and a bigger number of training data.

It puts in evidence a middle deviation between 2nd test architectures and Validation test ones not bigger than 6-7%. That deviation can be considered acceptable, together with errors percentage bad values for certain networks, because of the lost of training information. In the following schedule all results for architecture VD are shown.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>network</th>
<th>$SSE_{train}$</th>
<th>$SSE_{total}$</th>
<th>% $% &lt; 0.1$</th>
<th>% $&lt; 0.01$</th>
<th>% resp. corr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5,1</td>
<td>173,1</td>
<td>46,69</td>
<td>38,67</td>
<td>77,83</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>31,02</td>
<td>202,32</td>
<td>41,98</td>
<td>37,73</td>
<td>71,69</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>5,1</td>
<td>111,57</td>
<td>39,15</td>
<td>28,3</td>
<td>81,13</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5,1</td>
<td>193,91</td>
<td>41,98</td>
<td>36,32</td>
<td>85,84</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5,1</td>
<td>156,74</td>
<td>39,62</td>
<td>34,43</td>
<td>86,32</td>
<td></td>
</tr>
</tbody>
</table>

Figure 19 : results schedule for VD architecture

Multilayer architectures present better values of training SSE, which is due to the presence of two layers both with ten neurons which increase it capability to memorize. On the other hand, validation SSE values are not better than one layer architecture ones. It can be affirmed that simpler networks show a better generalizing capability than others as confirmed by VD maximum percentage of correct outputs, 86.39%.

To sum up, it has been demonstrated the effective simulation capability of the above described neural architectures in spite of the lack of information and the increasing of validation data, with satisfactory results.

**TESTS RESULTS AND POSSIBLE APPLICATIONS**

Previous chapters results have shown the possibility to design and use trained neural networks for human judgement simulation on Domus $K_4$ parameter. Except for architecture 3H, all results can be considered satisfactory and characterized by values that exceed 89% and in one case on three reach 98%. Therefore, generalizing capability and all results must be related to the described number of data and their distribution limits in order to implement such instruments into an effective maintenance system. It is normal, in fact, that a neural network trained with the first three of five output classes would probably produce wrong results for the last two classes, and especially for the fifth one, because it has no “experience” about them. Two different kinds of application can be found for neural networks in the field of visual inspections: decision making support and judgement validation. For the first application it would be convenient to formulate an empirical function depending on $K_4$ values of human and neural network judgements like Euclide’s distance and weighted average.

For the second one, a comparison algorithm could be implemented involving different actions according to a predetermined border value of the difference between human and neural network $K_4$.

**REFERENCES**

C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford-UK 1995;
Iztok Perus Jas Znidaric, *Deterioration Categorization and Priority Ranking by a Neural Network-Like Approach*, Z.A.G.-Slovenian National Building And Civil Engineering Institute, Ljubljana, Slovenia, CEB-Bulletin d’information N° 243 (1998V);
GRANULAR SYSTEM MODELS FOR REAL TIME SIMULATION

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@uv.es

KEYWORDS Cellular automata, real-time, simulation, sand pile model.

ABSTRACT
Granular systems appear in many industrial and agricultural disciplines, which has lead to a wide range of models for their simulation, both in engineering and applied physics. However, most of the current models are addressed to make a deep analysis of the properties of the system, and they are not suitable for their use in real-time simulations. Thus, despite of the efforts in computational fluid dynamics and molecular dynamics, the application of granular systems in the field of virtual reality and computer graphics lacks of a set of fast interactive models that allow their execution in real-time. In this paper we propose an efficient granular system model based on cellular automata, designed to be used in computer graphics applications. The model is provided with inertia which gives it more realistic physical properties. Also, several details for its implementation are given together with an analysis of its computational complexity.

INTRODUCTION
Granular systems can be found in nature and in industrial applications (desert dunes, concrete used in civil engineering, granulates employed for fertilizers, ...). Due to its complexity, the properties of their dynamics has been a widely studied problem that still remains open.

Continuous media mechanics was the first approach to the modelling of these systems that considers the systems as a kind of flow (de Barr Saint-Venant, 1850), and is still widely used (Aranson and Tsimiring, 2001). However, the behavior of a granular system is somewhere in between a flow and a solid, showing properties of both states (Hornbaker et al., 1997). During the seventies, and mainly thanks to the increases in computer power, a new simulation approach was developed (Cundall and Strack, 1979). In this approach every particle of the system is considered separately and the evolution of the system is computed from the interaction between all the particles and their trajectories according to newtonian Laws.

A more recent approach is taken in the models developed by Bak et al. in their work on Self Organized Criticality (Bak et al., 1988). It allows the study of some statistical properties of the systems. However the models presented by the different authors lack realism, at least in regards to the spatial behavior of the system (Christensen et al., 1992; Chen and den Nijs, 2002).

However, it has been only in the last years that complex systems models have been introduced in the computer graphics field, and almost exclusively applied to fluid dynamics simulation (Stam, 2003; Müller et al., 2003).

Pla-Castells set some basis for the development of cellular automata models suitable for simulation of granular systems in graphics applications (Pla-Castells et al., 2003). In this paper we present a cellular automata model suitable for real-time interactive simulation that improves and completes the ESPM model presented there.

The structure of this work is as follows. First the ESPM model is reviewed, and several improvements and modifications are presented, leading to a more complete granular system model. In the second section an inertia model is proposed to give more realism to the properties of the model. In the third section, the suitability of the new model to computer graphics applications is discussed and some implementation details are given.

A GRANULAR SYSTEM MODEL
In (Pla-Castells et al., 2003) it was presented the ESPM model; a granular system model based in the Sand Pile Model (Bak et al., 1988). That work extends Bak’s model to allow avalanches in every direction and uses a variation of the Abelian Sand Pile Model rule (Dhar, 1990), to fit the needs of real time computer graphics applications. Taking that model as the starting point we will introduce some modifications to improve its dynamics.

The ESPM model used a rectangular grid, divided in square cells, that represents the ground where the system is laying. Each cell \( (i, j) \) has a state variable \( h(i, j) \) that represents the height of the granular system in the center of the cell \( (x_i, y_j) \).
The dynamics of the model is achieved by applying a rule based on the Abelian Sand Pile Model cellular automata (Dhar, 1990). For every cell, a neighborhood is defined, commonly formed by the four or eight closest cells. The height difference for each pair of neighbor cells is computed; if this difference is greater than a threshold \( \delta_t \), the taller cell is reduced a quantity \( z_+ \), and the shortest cell is raised the same quantity. This process is repeated for each cell in the grid. In order to keep isotropy, the order in which the neighbors of a given cell are compared is taken randomly.

Algorithm 1 shows the pseudo-code that corresponds to this model. In this algorithm, \( n_v \) represents the number of neighbors of a cell. For each \( c \in \{ 1, \ldots, n_v \} \), \((a, b) = \text{neigh}(c)\) \( a, b \in \{-1, 1\} \) represents the relative coordinates of that neighbor respect to the current cell.

**Algorithm 1**

```plaintext
for i = 2 : N - 1
    for j = 2 : N - 1
        for n = 1 : n_v
            (k, l) = neigh(n);
            diff = M(i, j) - M(i + k, j + l);
            if (diff > \( \delta_t \))
                M(i, j) = M(i, j) - z_+;
                M(i + k, j + l) = M(i + k, j + l) + z_+;
            fi
            if (diff < -\( \delta_t \))
                M(i, j) = M(i, j) + z_+;
                M(i + k, j + l) = M(i + k, j + l) - z_+;
            fi
        end
    end
end
```

One of the main limitations of the ESPM model comes precisely from the fact that the neighbors are checked randomly. This allows situations in which a neighbor of a cell can remain unchanged during several steps, usually produced when the previous updates reduce the cell’s height difference below the threshold. That breaks the intuitive idea that the material would flow in the direction in which the slope is at a maximum, instead of being randomly spread, and leads to unnatural configurations.

In order to fix this drawback, when checking a cell, all its neighbors will be tested before any update is done. Then the updates will be done by order of difference magnitude, starting with the cell with the biggest difference.

Another limitation of the ESPM model is the fact that the start of an avalanche is decided by difference of height. In real granular system avalanches depend on the slope of the system. Thus, to properly compare two cells, the slope has to be used and the distance between them has to be taken in account. This is specially important if we use the eight element neighborhood, as the cells at the corners are further than those at the sides. We will use an estimation of the slope between two cells, obtained as

\[
\theta = \arctan(\delta_t / d_c)
\]

where \( d_c \) is the distance between the checked cells.

As a third main improvement we will review the size of the material transfers that take place when the model is updated. According to the rule defined in the ESPM, the quantity of material that moves at every update is fixed, and it is determined by the value of the parameter \( z_+ \). In the new model this value is variable and proportional to the slope between the cells: \( z_+ = \alpha \theta \). With this formulation the model leads to a greater flow when the slope is high, and to a smaller flow with low slopes.

Algorithm 2 presents the pseudo-code of the new granular system model. In this algorithm, the routine \((m, slo) = \text{get_max_slope}()\) computes the maximum slope between the cell \((i, j)\) and its neighbors, giving as output this slope and the index of the neighbor at which it occurs.

**Algorithm 2**

```plaintext
for i = 2 : N - 1
    for j = 2 : N - 1
        (m, slo) = get_max_slope();
        (k, l) = neigh(m);
        if (slo > \( \delta_t \))
            M(i, j) = M(i, j) - \( \alpha \cdot slo \);
            M(i + k, j + l) = M(i + k, j + l) + \( \alpha \cdot slo \);
        else
            M(i, j) = M(i, j) + \( \alpha \cdot slo \);
            M(i + k, j + l) = M(i + k, j + l) - \( \alpha \cdot slo \);
        fi
    end
end
```

**MODEL WITH INERTIA**

According to mostly works on granular systems, a realistic behavior of cellular automata models is achieved only when inertia in the matter flow is considered (Held et al., 1990; Goles et al., 1998). The effect of inertia in granular systems confer them some characteristics that are not observed in models without inertia; the slope of the mountain formed by the addition of material in one point follows a logarithmic curve, instead of a straight line (Alonso and Herrman, 1996); and the behavior of the avalanche frequencies that have a transition from a 1/f noise distribution to a spectrum determined by the size of the mountain (Prado and Olami, 1992).

When considering inertia in the model, it must be taken into account that the conditions that make the material flow are more restrictive at the beginning than when the material is already moving. The model of inertia that we introduce here has the advantage of allowing an implementation that does
not require the storage of any additional information besides the state of the system.

The effect of inertia is achieved by introducing a second threshold $\delta_I < \delta$ that we shall name inertia threshold. When a pair of cells are updated, the cell that increases height of it is checked to determine whether it can be updated again according to the height at its neighbors, but using the inertia threshold $\delta_I$ instead of the original threshold $\delta$. If that is the case, an amount of matter $\alpha_I \theta$, with $\alpha_I < \alpha$ would be transferred in this new update.

The effect of this addition to the model is that the critical angle of the system is lower in a place where the material is flowing than in a place where the material is resting. This leads to a smaller slope in the lower part of the pile, where the avalanches end, giving it the expected logarithmic correction. The difference between the original ESPM model and the new model with inertia can be seen in Figure 1.

![Figure 1: A system formed by deposition on one point. Comparison between the ESPM model and the new proposed model (below).](image)

**REAL-TIME REPRESENTATION OF A GRANULAR SYSTEM**

The aim of the model presented here is to allow the simulation of Granular Systems in Real-Time Interactive Computer Graphics simulations. Due to its complexity, the problem of visual representation of fluid systems is an active research field, with many recent publications related to gas, dust, liquid or textile animation models (Carlson et al., 2002; Müller et al., 2003; Baraff et al., 2003; Stam, 2003).

The case of granular systems inherits the drawbacks of this complexity, as mostly models come from these fields. On the one hand, visual representation of models based on fluid mechanics, commonly involves the identification of the surface of the system from the model state variables and a proper discretization (Foster and Fedkiw, 2001; Enright et al., 2002). On the other hand, molecular dynamics models suffer the problem that drawing the complete system is not affordable in real time, due to the high number of bodies that are considered. Recently some works have been published to identify the particles that define the surface of the system (Müller et al., 2003). However, in such works, the number of particles considered is much lower than the number we would need for the kind of system we are treating, and even in that case, the results cannot be computed in real time (Rasmussen et al., 2003).

Following we shall discuss the suitability of the presented model for real time computer graphics applications. Firstly we will give some notes on how can our Granular System model be represented. We have chosen OpenGL, since it is one of the most commonly used graphic libraries (Neider et al., 1999).

The state of the system is stored in a matrix $H$ in which the value at position $H_{ij}$ represents the height at the cell $(i, j)$. Thus, the most suitable representation for this kind of data will be a manifold that represents the surface of the system. This surface will actually be an interpolation of the set of points

$$G := \{(x_i, y_j, h(i, j))\}.$$

This representation can be done in a very straightforward way using the OpenGL primitives QUAD_STRIP and TRIANGLE_STRIP. A QUAD_STRIP draws a group of connected quadrilaterals, defined by a list of points $\{x_i\}_{i=1}^{2N}$. The quadrilateral $n$ is defined by points $2n - 1$, $2n$, $2n + 1$ and $2n + 1$. In the same way, a TRIANGLE_STRIP draws a group of connected triangles, defined by a list of points $\{x_i\}_{i=1}^{N}$. If $n$ is odd, triangle $n$ is defined by vertices $n$, $n + 1$, and $n + 2$, and if $n$ is even, triangle $n$ is defined by vertices $n + 1$, $n$, and $n + 2$. Both primitives are shown in Figure 2.

These two structures need that position of every vertex to be set. In the model we present, all the information needed to set these points is the values of the cells of the automata. Thus, representing the system with the cellular automata model using OpenGL can be done by a simple evaluation of the matrix. Figure 3 shows an OpenGL image of a simulation of a granular system using the proposed model.

**Complexity of the model**

According to the method proposed to represent the system, the use of the model implies the computation of $N^2$ tests to update an $N \times N$ automata. This burden can be
too high for most real-time applications. However, a cellular automata admits some optimization that can be used in this case. A record of the cells that were updated is logged at every step. Then, at the next step only those cells that changed are computed. Exhaustive tests have been performed to obtain the cost of the model in a system formed by deposition of material at one pint (Pla-Castells, 2003). The results have shown that the number of cells tested in a step is in linear relation with the diameter of the granular system (see Figure 4). Thus, in any case, the maximum cost has been shown to be linear in relation with the side of the grid of the cellular automata.

### Normal direction computation

In order to obtain a proper representation with OpenGL, a normal vector has to be also defined at every vertex of the mesh. This vector is used by the illumination algorithm of the scene renderer. In the presented model, this vector can be obtained with no additional calculus that those needed to update the state of the automata. Let us consider the surface of the sand pile as a smooth surface defined through a $C^1$ function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, as the set $S := \{ (x, y, f(x, y)) \}$, and the set of points defined by the automata

$G := \{ (x_i, y_j, h(i, j)) \},$

as a discretization of this surface. According to the definition of the normal vector to a smooth surface and if we approximate the derivative of a function by means of differences, we can compute the normal at every vertex as

$$N(x_i, x_j, h(i, j)) = \frac{t_1(i, j) \times t_2(i, j)}{||t_1(i, j) \times t_2(i, j)||}$$

where

$$t_1(i, j) = \left(1, 0, \frac{h(i + 1, j) - h(i, j)}{x_{i+1} - x_i}\right)$$

$$t_2(i, j) = \left(0, 1, \frac{h(i, j + 1) - h(i, j)}{y_{j+1} - y_j}\right)$$

The most important is that all the differences in $h$ needed to obtain this vector have been previously computed to check the evolution of the cellular automata, and the differences $d_1 = x_{i+1} - x_i$ and $d_2 = y_{j+1} - y_j$ are constant values given by the size of the automata and the number of cells. Therefore, no overhead is added in the normal calculations.

These results make the proposed model to be a high performance model, that widely outperforms the other modelling approaches made with graphic representation purposes.

### CONCLUSIONS

In this paper we have addressed the development of a model of granular systems that can be applied in real time interactive computer graphics applications. The model presented shows the main characteristics of real granular systems, such as the angle of response, the appearance of
avalanches and the reaction to interactions. Furthermore it has a very high computational performance.

We have also studied some details of the implementation of the model and of its use in one of the most widely used computer graphics tools, the library OpenGL, showing that the model not only is efficient, but also has a straightforward application to the field to which it is aimed.

The use of the proposed model in interactive applications makes it necessary to develop a set of interaction models that allow the computation of the response of the system under different perturbations (Lifiman et al., 2001; Bagi, 2003; Edwards and Grinev, 2003). As future work, some interaction models have to be developed.

Also, it would be desirable to develop a set of routines and an adequate programming interface that allow an easy use of the models in more complex programs.

REFERENCES


SIMULATION IN DESIGN
MARPET: MULTI-SENSORY SIMULATOR FOR ASSESSMENT OF VIRTUAL RAPID PROTOTYPES IN DENTISTRY INSTRUMENTATION.

Francisco Serrano Arrones, Antonio Díaz Estrella
Departamento Tecnología Electrónica. Universidad de Málaga.
España
Campus de Teatinos 29071 Málaga, SPAIN
ade@dte.uma.es

KEYWORDS
Haptic, rapid prototypes, force feedback, simulators.

ABSTRACT
This paper presents a multi-sensory simulator for assessment of dentistry instrument prototypes. The novelty is in the utilization of a force feedback device which allows to make tests to virtual dental instrument models. This facilitates a collaborative design to improve the quality and the development time.

1. INTRODUCTION
Since its birth virtual reality has attracted the curiosity of everybody. Virtual reality systems immerse us in synthetic worlds presenting stimuli to our senses and achieving sometimes spectacular results. Sight, hearing and touch are the human senses that provide the bigger part of information we receive. This is one of the reason of the appearance of haptic devices.

Haptic is defined as related or based on the touch sense. Haptic devices provide an immersion in which the virtual objects are not only watched and heared but also touched, this opens a stunning field of possibilities in industry, entertainment, scientific research etc. similar to the field opened in its moment by the ‘traditional systems’ of virtual reality (head mounted displays and gloves). Today there are numerous haptic devices. They differ on the corporal zones they affect (fingers, hands, arms etc.), or on the technology they use (exoskeletons, force feedback, tactil devices, point sources, specific task devices, etc.). (Fish et al)

Although these devices are a novelty for the society on the whole, at industrial level they are known and used since, even, the sixties. The nuclear industry handles equipments that functionally are really similar to the current desktop haptic feedback systems. These tools are the Master-Slave Manipulators (MSMs) for safe remote handling of irradiated material under direct human control.

Entertainment is one of the industries that are benefiting from haptic devices and possibly will continue doing so in the future. Since five or six years ago it has offered the public several affordable haptic devices. A joystick, a mouse and a steering wheel with force feedback are some of the systems they market. The industry of manufactured product is another field of interest. A good example is the FreeForm Concept system from Sensable Technologies. FreeForm, which includes the PHANTOM Omni haptic device, allows designers to quickly rough out models and render them to test out different design directions before committing to detailed models or physical prototypes. This is very interesting for rapid prototyping.

One of the virtual reality traditional applications is that of training within a simulated environment. The reasons for this area are: because they are dangerous for the operators life (i.e: delicate operations in nuclear power station), or because the devices are really expensive (i.e: flight simulator) or if the training is necessary because the life of other person is at stake (medical operation). Again, Industry has a good opportunity to benefit if it invests in haptic systems to simulate this kind of process.

2.- PHANTOM DEVICE.
The Phantom is an electromecanical force feedback device which belongs to the category of ‘point source’. It is handled by introducing a finger in a thimble or taking a stylus. The interaction is produced on the tip of them, so we apply forces to this point and receive some forces back, giving the impression the object is being touched.

The Phantom has a configuration with several interesting advantages. Its programation is easy, due to it only has one contact point and provides a greater fidelity. Besides this, its size allows an easy use inside an aceptable work space. Lastly, say that its structure enables that the programmer can define the pen or thimble as a medical tool in a surgery application, a brush in a 3D paint application etc. This gives a lot of flexibility.

A lot of projects have been developed based on the technology that Phantom offers. In medicine is specially indicated for minimally invasive surgery, laparoscopic or arthroscopy – Boston Dynamics Inc. and University of Colorado Health Center are examples of companies and universities that work with this technology. Another of the most important applications is the previously commented application destined to the industrial production of ceramics and manufactured products in general. Nowadays companies such as Wedgewood and Royal Doulton have invested heavily in this area so as to increase and improve their production using this application device.
The aim of the work is the development of a friendly and realistic haptic interface to test three-dimensional models of the tools applied on dental operations.

3.- MARPET.

3.1.- System Description.

The development platform used is a Reachin Display from Reachin Technologies. Reachin Display combines a stereo-visual display and a Sensable’s Phantom haptic device. A semi-transparent mirror creates an interface where the user can see and feel the virtual object in the same place. In figure 1 is shown a Reachin device photo.

![Figure 1: Reachin Display.](image)

In a first step a review of the developed researches in the field of virtual environments design for dentistry was made. This provided several ideas to program the application presented in this paper. (Wang et al. 2003) (Chen et al. 2003)

The multi-sensory simulator creates a natural environment for the dentist that includes:

a) Dentist’s instrument prototypes. The medical instruments can easily be designed under a CAD application and immediately introduced in the scene where the dentist can evaluate it.

b) A virtual life-size head (patient). The patient seems to be real, he makes expressions, his skin is soft and deformable and it is possible to feel the difference between several tissues. The head can open its mouth and show the teeth which are designed following human denture structure.

c) Study cases for assessment: Tooth Decay, tartar and pain.

d) A friendly tactile interface for configuration (patient, instruments, cases, etc.) and control (see figure 2). With it you can choose and detail the dentistry prototypes to be tested. It also allows the user to decide between a scene with one tooth or a another where it is possible to choose the ailment of each tooth of the patients mouth.

The dentist can manipulate the virtual instrument and carry out the different tests for assessment which are based on observable criterions like ergonomics, size, and mobility inside the mouth. The dentist can take advantage of the stereoscopic vision and haptic properties of the system to evaluate the prototypes in a scene similar to that of the real environment without the physical existence of them.

![Figure 2: Tactile Interface.](image)

With this simple approximation it is possible to perform collaborative design in which the three-dimensional model is developed by a specialist in 3D design while the test in the virtual environment is done by the dentist.

3.2.- Test.

The ergonomic of hand instruments is needed not only for the comfort of the patients and a better quality of operations but so as to avoid injuries in the hands and the wrists of the dentists. Figure 3 shows the environment where the assessment process is made.

![Figure 3: Assessment Scene.](image)

Characteristics such as sharpness of cutting edge, weight, balance, diameter, size etc. bring about the ergonomic of dentistry instruments. Some of them, such as sharpness, diameter, size can be simulated in this application. Others such as geometry or material can be controlled also.

To avoid dentist tiredness all the kind of instruments models and shapes can be previously tested. Even several designs can be quickly assessed because the same kind of device could have several different versions. These may vary in material or geometry depending on the dental zone where they are operating on. The user position and the scene which is presented by the application during the assessment process is shown in Figure 4.
3.3.- Dental Models.

The different dental models will be useful for testing characteristics such as the type of material, velocity of rotation and instrument geometry.

The best way for these type of applications are volumetric dental models with several layers which may be updated as the medical tools work on them. In this way it is easy to program the different sensations within the teeth tissues.

In a first approximation these structures are not used, but instead an empty tooth defined by a surface. With the target of achieving volume sensation three structures have been developed. The way in which the depth sensation is actualized is acceptably realized.

The first one imitates a tooth decay where the localization and size is aleatory. The action of the virtual burr pushes the teeth vertex and actualizes the affected zone so the tooth decay disappears as it makes it deeper. In this model the action can be controlled with coefficients or with the applied force. Its implementation is interesting because it has developed a surface which transforms rigid objects to malleable when it is stuck to them. The next step will be to accomplish elasticity deformations.

The creation of the tartar requires two layers: superficial and inner. The superficial one is a tooth decay previously treated and the inner is a virtual rigid tooth. This treatment consist of cutting a part of the tooth out (this is due to the tartar only appearing on the tooth base), changing the texture (to achieve realism) and a scaled (1.001 bigger, so that it creates the upper layer).

The last process imitated is the reproduction of pain in some tooth through applying heat-cold. To realize this it was enough to define aleatory zones which were programmed to be sensitive to these sensations.

3.4.- Conclusions.

The final application represents a really valid tool for the dentistry instrument industry because they will be able to test products in a virtual environment, in situations very similar to that of the reality environment before they mass produce them.

It promotes an efficiently collaborative design of prototypes due to the designer and the tester (dentist) work together. The first can vary the prototypes characterists and the second checks the results in the moment.

Finally, due to its characteristics, the system can be also used as a dentist training application. Currently there are companies, such as Novint Technologies, that sell relative similar applications for the education within dentistry faculties.

3.5.- Future Work.

It is necessary to succeed getting virtual three-dimensional dental models that corresponds with reality, to achieve this, it will not be used superficial approximations. Then volumetric structures where the penetration or fracture in the teeth corresponds with the disappearance of layers in the virtual models will replace the currently models

Presently there are several studies and master thesis which exist to provide some interaction models between teeth and instruments. (Agus et al. 2003) (Sae-Kee et al. 2004)

Accurately the last reference explains a method to represent the previous interaction which will be used to a possible actualization because it permits to control variables as the force applied, angular velocity of the burr, geometry, material etc. To design good dentistry instruments.

REFERENCES.


VIRTUAL WORKSHOP FOR MILLING PROCESS SIMULATION

Haslina Arshad, Che Hassan Che Haron¹, Syaimak Abdul Shukor, Shahrul Azman
Mohd Noah⁵, Rosilah Hassan¹, Zainal Rasyid Mahayuddin

Department of Industrial Computing,
Department of Computer Science²,
Department of Information Science³
Faculty of Technology and Information Science,
Universiti Kebangsaan Malaysia,
43600 Bangi, Selangor, Malaysia.
Tel:+603-89216722, Fax:+603-89256732
Email: has@ftsm.ukm.my

Department of Mechanical and Material Engineering¹,
Faculty of Engineering,
Universiti Kebangsaan Malaysia,
43600 Bangi, Selangor, Malaysia.
Tel:+603-89216516,
Email: chase@eng.ukm.my

KEYWORDS
Milling process, Virtual Reality

ABSTRACT

Manufacturing education on a topic such as manufacturing process is traditionally done in physical laboratories and operation of complicated machines requires special training. Virtual Reality has been proposed as a new training tool, which will benefit college students, and manufacturing workers through its computer generated virtual environment. In this work, a Virtual Workshop is designed to aid students in learning about the milling process and get trained in operating the machine virtually. The 3-D simulation models are developed using AutoCAD 2000 where they will then be transformed into VRML format. Meanwhile, JavaScript is used in developing the animation for the milling process simulation. As an interactive virtual workshop, it also provides relevant information on Computer Numerical Control (CNC) milling machine and also information on how to perform the milling process efficiently. This will also help university instructors to illustrate the whole concept of milling operation.

INTRODUCTION

Virtual Reality’s concept has been implemented in this work to provide three-dimensional world for milling process. Virtual reality, also called virtual environments, is a new interface paradigm that uses computers and human-computer interfaces to create the effect of a three dimensional world in which the user interacts directly with virtual objects. The term virtual reality is defined as a way for humans to visualize, manipulate and interact with computers and extremely complex data (Aukstkalns et al. 1992). The visualization part refers to the computer generating visual, auditory and other sensual output to the user of a world like a scientific simulation or CAD model within the computer. The user can interact with the world and directly manipulate objects within the world. Therefore, virtual reality is regarded as computer generated, three dimensional and interactive (Bryson, 1996).

Manufacturing education on a topic such as manufacturing process is traditionally done in physical laboratories and operation of complicated machines requires special training. Virtual Reality has been proposed as a new training tool, which will benefit college students, and manufacturing workers through its computer generated virtual environment. In this work, a Virtual Workshop is designed to aid students in learning about the milling process and get trained in operating the machine virtually. Virtual Workshop represents a module of virtual manufacturing system. A virtual manufacturing system is an integrated, computer-based model that represents the physical and logical behavior of a real manufacturing system (Ong et al. 2002).
Milling Process

Milling process is classified as a material removal process. This process and its machine tools are capable of producing complex shapes such as stepped cavity or tapped hole with the use of multifunctional, as well as single point cutting tools. In the milling process, a multifunctional cutter rotates along various axes with respect to the workpiece. Some of the basic types of milling cutters and milling operations are slab milling, face milling and end milling as displayed in Figure 1. In slab milling, also called peripheral milling, the axis of cutter rotation is parallel to the workpiece surface to be machined. The cutter, generally made of high-speed steel, has a number of teeth along its circumference, each tooth acting like a single point cutting tool called a plain mill. Meanwhile, in face milling, the cutter is mounted on a spindle having an axis of rotation perpendicular to the workpiece surface. End milling can produce flat surface as well as various profiles. The cutter in end milling has straight or tapered shanks for smaller and larger cutter sizes. The cutter usually rotates on an axis perpendicular to the workpiece, although it can be tilted to machine tapered surfaces (Kalpakjian et al. 2003).

![Figure 1: Schematic illustration of a milling operation](image)

Source: (Kalpakjian et al. 2001)

Related Work

Nowadays, numbers of applications of VRML exist on the Web in various areas. One of them is a method for simulating basic manufacturing operations such as unload, load, process, move and store in a 3D virtual environment. The virtual environment provides a framework for representing a facility layout in 3D that consists of the static and dynamic behavior of the manufacturing system (Chawla, R & Banerjee, A, 2001). The application of VR in other field such as NC machining simulation has been developed and this prototype system employs Java, external authoring system (EAI) and virtual reality modeling language (VRML) to perform the simulation (Qiu et al. 2001).

Another invention in VR application is a virtual machining laboratory for knowledge learning and skill training in an interactive environment. This virtual lab is specifically designed for helping students to virtually operate a lathe or set machining parameters and input CNC G-code program to turn the work piece automatically (Fang et al. 1998). Meanwhile, Delmia’s Virtual NC is also an interactive 3D simulation environment for visualizing and analyzing the functionality of an NC machine tool, its CNC controller and the material removal process (Delmia 2001).

A web approach was done in building a CRT-based virtual reality visualization tool to allow literacy students to take virtual tours of various hardware components (Watterson, 2001). The tool is built under the concept of integrating VRML, JAVA, and HTML. The tool has been used in several literacy classes and resulted in students expressing excitement and enthusiasm about the tool and learning how to navigate in a virtual world.

CONSTRUCTING THE VIRTUAL WORKSHOP

In developing the application, there are three important stages and they are clearly illustrated in Figure 2.

![Figure 2: Flowchart of the developing stages](image)

Authoring 3D world in AutoCAD

The three dimensional model was first created using AutoCAD 2000. To form a 3D world of a milling process, only selected parts of the CNC machine that are involved in the process are created instead of developing a complete structure of a typical CNC machine. The drawing is just an illustration for educational purpose; therefore, the main parts are drawn based on a typical CNC machine where the scales and structures are not accurate and should not be compared to the real machine. Initially, the three
dimensional model of CNC Milling Machine were drawn in a wire frame form before the solid model of the machine is displayed (Figure 3).

![Figure 3: Authoring 3D model in AutoCAD](image)

When the authorizing stage is completed, next comes the translation process where a group of objects that represents the components of the machine is being exported to the VRML format (*.wrl) from the AutoCAD 2000 (*.dxf) format. For example, the worktable would be exported together with the T-slots that are joined with the worktable itself as one group. This procedure is repeated for other groups until the entire model has been translated to the VRML format. The entire translation process is done by the VRML Out (see Figure 4), a software that was released by AutoDesk to allow the exporting of any 3D solid model from AutoCAD format to VRML format.

![Figure 4: Exporting 3D model with VRML Out](image)

After that, all the groups of file are gathered once again using a VRML editor, VRML Pad, to form a complete VRML model as previously seen in the AutoCAD drawing. Now, the virtual milling process world is done. It is ready to be explored through any VRML browser but it is yet to be equipped with the multimedia effects.

**Coding**

Before this, the virtual world looks static. The coding part is carried out in order to equip the virtual world with multimedia effects such as animation, the color of the materials, sound, lighting and setting the appropriate viewpoints (the way the user view the world). The example codes for creating the sound of door’s creak is shown in Figure 5.

```javascript
DEF Sound Sound {
    Source DEF Creak Audio Clip {
        description "door_creak"
        start_time 0
        url "/..../Sound/door.wav"
    }
}
```

**Figure 5: Example codes of VRML**

**JavaScript**

JavaScript is used to examine any situation and make the appropriate action, for example to produce the animation of a door that is initially closed by clicking on it. Assuming that in the beginning the door is closed, clicking the door would cause certain translation (positive to x-axis) at the door, so the door is now opened. At this moment when we click on the door again, it would cause the same value of translation as before but it is at the opposite direction (negative to x-axis). Therefore JavaScript could decide which action has to be taken at the appropriate moment. After completing all the authorizing and coding, the virtual world would now become more flexible, interactive, attractive and much resembles our real world. Figure 6 shows the example codes of JavaScript for operating the door.
Open the door
DEF OPDdoorscript Script {
  eventIn SFTime touchTime
  field SFIInt32 Start 0
  field SFIInt32 Stop 1
  field MFVec3f tra1 [1.0 0.0 0.0, 20.0 0.0 0.0, 30.0 0.0 0.0]
  field MFVec3f tra2 [1.0 0.0 0.0, 0.0 0.0 0.0, 0.0 0.0 0.0]
  eventOut MFVec3f outtra
  url "javascript:
    function initialize() {}
    function touchTime(value, time) {
      if(Stop==1){
        Start=1; Stop=0;
        outtra=tra1;
      } else{
        Stop=1; Start=0;
        outtra=tra2;
      }
    }
}"

Figure 6: Example codes of JavaScript for operating the door

Perform the application in web page

Locating the completed vrml file (Virtual Milling process world) into the web browser is the final stage of the project development. The main page contains some useful reading material about the milling process, CNC machine, introduction to Virtual Reality and of course the link to bring the user to the virtual workshop.

PROTOTYPE OF VIRTUAL WORKSHOP

The interface of the virtual workshop is a page that briefly explains the project as shown in Figure 7. This interface has four main buttons such as Home, Introduction, Milling Equipment and Milling Processes, which has some description about the virtual workshop and the milling process.

For the submenu of milling process, selecting each title would display the virtual reality world respectively on the target frame with the appearance of Cortona (VRML browser) menu bar on the left and bottom of the frame. The virtual world displayed under Cortona could function normally and could be explored freely. The model could be rotated randomly. The user can move closer or further from the object.

This virtual workshop is capable of displaying three milling processes such as slab milling, face milling and end milling. For each milling process, it has three buttons to control the animation during the geometrical simulation such as start, stop and sound as shown in Figure 9.

For better visualization, the user can click the full screen button. Here, the finish product for each milling process, the cutter and the milling process will be displayed to the user as shown in Figure 10.
DISCUSSIONS

The interactive concept of Virtual Reality application has succeeded in making it closer to reality. The VR concept that allows exploration throughout the 3D world enables the user to be in a world where they can move freely, nearer and further relatively to an object, being able to rotate an object, changing the viewpoint and many other interactive features.

It is considered as a new and attractive method of learning or information presenting. The user need not be in the CNC lab or certain workshop just for the reason of learning the milling process. Much time would be wasted and it will cost some money in operating the CNC machine. Through this application, the safety issues were being considered as well where the user would only have to deal with the computer screen while observing the process rather than the real machine that require certain skills in operating it. This virtual workshop is the best medium to introduce one of the significant manufacturing processes, milling process to those people who do not have any technical background on this area.

CONCLUSION

Virtual Workshop For Milling Process Simulation has been successfully developed. This application, which simulates the milling process virtually, allows the user to explore through its world. It has provided a new teaching tool and useful information on the milling process and the related parts of the CNC machine to the user.

REFERENCES


DEVELOPMENT ENVIRONMENT FOR DESIGN OF 3D APPLICATIONS WITH FORCE FEEDBACK

Ernesto Jesús de la Rubia-Cuestas
Antonio Díaz-Estrella
Dpto. Tecnología Electrónica, Universidad de Málaga
Campus de Teatinos 29071 Málaga SPAIN
e-mail: ade@die.uma.es

KEYWORDS
Development Environment, VRML, Reachin, Virtual Reality, 3D Interfaces, Force Feedback.

ABSTRACT

In this paper, a new VRML based environment is described for designing virtual reality scenes and creating applications with 3D interfaces and force feedback. The developed applications run on a virtual reality system called Reachin Display. This system provides stereoscopic vision and force feedback. Reachin Display is a new product, which is in a development stage, and as of yet, there are no specific tools for the design of virtual environments for this system. Running applications in a 3D environment has many benefits, among them, working in a natural way and a reduction in development time. These 3D applications can be used in education, training, simulations, creation of development tools, etc.

1. INTRODUCTION

In recent years new technologies have changed the world. They are applied in all kinds of fields: business, industry, home, etc. And the current trend is that these new technologies development continue to grow. In this way, it is really difficult to imagine how, in short term, the future will unfold.

Along with the development of the new technologies, virtual reality (VR) techniques play a more and more important role. It is expected that, in the next few years, virtual reality techniques will appear in all kinds of environments. They will be applied to education, simulations, leisure, training, industry, etc.

Among the most important characteristics in a virtual reality system are stereoscopic vision and force feedback. With current technology, it is possible to get satisfactory stereoscopic vision, however, providing a realistic force feedback is still a difficult task. The current solution for this problem is to restrict it to particular situations where the problem is easier to solve.

A system that uses this partial solution is Reachin Display from Reachin Technologies which provides stereoscopic vision using wireless shutter glasses and force feedback based on Sensable's Phantom (Poling et al. 2003), Reachin has a metallic structure which holds the monitor in an inclined position, so that the user can see the virtual scene reflected in a mirror. In this manner, the user can work in a natural and comfortable position (see figure 1).

Figure 1: Reachin Display

This general approach makes it possible to find a lot of useful applications for this system (Laycock et al. 2003). Reachin Display is used in medicine as a training simulator for surgical operations. It is also used in the industry as a simulator system, for example in welding simulators. Other industrial applications are 3D prototype design, and 3D data visualization. All these applications reduce the development time using the stereoscopic vision and tactile interaction, which allows the users to work in a natural way.

Reachin Display is a product in the development stage. The number of its 3D applications is increasing every day. It is one of the first steps in the evolution of interfaces from 2D to 3D.
2. DEVELOPMENT ENVIRONMENT

The 3D scenes used by Reachin Display, are based on the VRML97 language (The Virtual Reality Modelling Language – ISO/IEC 14772-1). This language was designed to create 3D web pages. Reachin Display uses a modified version of VRML. This new version adds information about how the user feels the virtual objects when they are touched by the phantom device.

This modified version of VRML is not fully compatible with the standard VRML 97. Therefore, all the existing 3D design tools cannot be used directly to work with Reachin Display. For the moment, the only way to work with the VRML modified version of Reachin Display is to use a text editor.

This situation has provoked us to create a development environment to design virtual scenes and 3D applications, which is able to work with both file types, VRML 97 standard and the VRML Reachin Display version.

This development environment makes it easier to work with these file types providing two independent edition modes: Tree and Text.

In Tree edition, it is possible to create a tree structure with the elements of the VRML language. This mode avoids the need to type the VRML by hand. In this manner, it is not necessary to remember the exact VRML syntax. Furthermore, the user can modify all data contained in these VRML elements with a friendly visual interface. Figure 2 shows an example of this interface.

![Figure 2: Tree Edition Mode](image)

The text mode allows the user to edit manually VRML code. To achieve this feature, the development environment provides a multi-file editor with several edition facilities. One of them is the VRML code auto-fill in text feature, which frees the user from having to remember the entire VRML syntax. In addition, the user does not need to type all the file code. This makes possible to create VRML scenes quickly and easily.

Another significant feature of the multi-file editor is the format VRML parser which recognizes all the VRML elements in the code and represents each one with a different appearance. This allows the user to understand the VRML code quickly. In addition, when the user types a reserved word of the VRML code, the change in its format confirms to the user that the typed text is correct.

This development environment is, in its two edition modes, very similar to a programming environment. It is possible to execute the VRML scenes without the need to save the code to a file and when an error is detected in a line, it is possible to edit this line with a simple double-click.

The VRML code can include scripts. This makes it possible to create 3D applications for Reachin Display. These scripts allow the implementation of complex processes such as working with files or accessing Internet with sockets. The text edition mode allows the user to create scripts, which can be executed with a simple click. If the user tries to execute a script code containing errors, the number of the error lines will be displayed. The user can edit these lines with a double-click.

Furthermore, the modified Reachin Display version of VRML, provides a set of elements to create 3D interfaces with force feedback. Examples of these 3D elements are buttons, slider bars, menus, etc. The development environment includes a utility to design these 3D interfaces. The user can use the Phantom device to place the controls in a quick and visual way in the desired position.

Another feature of this development environment is the file format conversion between VRML 97 standard code and the Reachin Display modified version. This option is very useful because it makes it possible to use all the existing 3D design tools to create VRML files, which can be used in Reachin Display. Visa versa, an object created with Reachin Display can be exported to other programs. All these features make this environment a helpful development tool for creating 3D applications with force feedback easily and quickly. This fact, along with the lack of this kind of application and the great amount of fields where Reachin Display can be applied, make this development environment a useful tool to create 3D industrial applications.

3. APPLICATION EXAMPLES

The next two examples, created with the development environment, illustrate all the previous ideas. These examples use script code, 3D interfaces with force...
feedback and include 3D objects designed with VRML97 tools.

3.1 Air-Traffic Controller 3D Interface

The current interfaces used by the air traffic controllers are 2D interfaces, which represent a 3D environment. There therefore exists the necessity to improve these 2D interfaces. Achieving a natural perception of these 3D scenes is desirable. In this sense, it is possible to use Reachin Display as a 3D air traffic controller interface.

In this example it is possible to visualize an airport, its airspace, and the planes, which are waiting to land. Stereoscopic shutter glasses provide 3D vision to the air-traffic controller. In addition, the Phantom device is used as a tracker, which allows the user to trace the path for the planes. The air-traffic controller can see in real time the evolution of the planes and can modify the path of each one if necessary. In this way the air-traffic controller determines the order in which the planes will land.

The 3D vision has two advantages. One of them is that the user can perceive, in a natural way, the exact position of all the planes. This could improve the alert state of the controllers and lessen fatigue and, as a consequence, reduce the risk of collision. The second advantage is that this simultaneous perception of the positions will make it easier to take decisions about the operations. Consequently, this 3D interface could help the air-traffic controller to make more efficient decisions. Figure 3 illustrates the appearance of this application example.

3.2 3D Industrial Design

Nowadays, tools used to design 3D objects are based in 2D interfaces, it is necessary to improve this situation. In this way, it is possible to use 3D interfaces (Gregory et al. 2000).

In this application example, the user can draw 3D lines choosing their color and the width of each one. These lines can be combined with any kind of 3D object. Another feature of this application example is to change the colour of the loaded 3D objects. This fact makes it possible to design industrial objects and export the prototype files to other applications.

The main advantage of this working environment is the reduction in development time. This is possible due to the fact that the user works in a 3D virtual environment in a natural way. Furthermore, handling this 3D application is easier than an equivalent 3D design program which uses a 2D interface. This offers another benefit, the reduction in the user’s training period. Figure 4 illustrates the appearance of this application example.

![Figure 3: Air-Traffic Controller 3D Interface](image)

**Figure 3: Air-Traffic Controller 3D Interface**

**4. CONCLUSIONS**

This paper presents a development environment for designing virtual reality scenes and creating applications with stereo-visual interfaces and force feedback based on Reachin Display. This tool provides a more natural interface for tasks that requires high attention and manual skills and also allows the reduction of development time in rapid prototyping.

**REFERENCES**


**ERNESTO DE LA RUBIA** was born in Málaga, Spain, and went to the University of Málaga where he studied telecommunications engineering and obtained his degrees in 2004. He begins to work in virtual reality in his end of degree project which is presented in this paper.

107
CONTINUOUS MOTION TRAJECTORY GENERATION FOR 3D SIMULATION

H. Chen, T. R. Wan and R.A. Earnshaw
EIMC, School of Informatics,
University of Bradford,
Bradford, West Yorkshire, UK
BD7 1DP

KEYWORDS
Continuous curvature curve, motion trajectory, trajectory generation, aircraft simulation, 3D simulation.

ABSTRACT
This paper discusses the motion trajectory generation problem in a three-dimensional (3D) space. We propose a continuous motion trajectory generation algorithm (CMTG), which can be implemented in real time following an arbitrary path defined by a number of dynamically allocated points in a 3D space. Previous works have mainly been focused on the static trajectory generation problem and on finding the optimised planar solutions for 2D applications. Our approach therefore offers a more flexible solution for real-time 3D trajectory generation problems. A number of simulation experiments have been conducted in the current work for evaluating the algorithm, and a 3D aircraft simulation has also been successfully implemented, which are described in the paper.

INTRODUCTION
Motion trajectory generation has received considerable attention in recent years. The main issue in this area is to generate a smooth trajectory curve joining two distinct configurations in the space with some constraints using interpolated piecewise polynomials. Curves have been on one hand an object of mathematical studies and on the other hand a tool for solving technical problems and other applications. There are at least two areas of curve applications, which deal with the construction of smooth curves. The first one of them is of the trajectory generation for moving agent path planning and motion control. For example, the simulation of an autonomous car or aircraft in a virtual environment, and the robot motion trajectory control in the real world. A C^2 smooth curve is necessary for the agent or its control system to track either in a virtual environment or in the practical world. The second application area is concerned with geometric design and modelling. For examples, curve design and creation on specific surfaces, and shape design in CAD applications. In this paper, we will focus on the motion trajectory generation and implementation in a 3D space.

Motion trajectory problem was early discussed by Dubins (1957), who proposed the solution using straight segments connected with tangential circular arcs of minimum radius and proved that the shortest distance between two configurations was such a path. However it is important to note that the curvature along such a trajectory curve is discontinuous; the discontinuities occur at the transitions between segments and arcs. The non-continuous curvatures may result in agent control difficulty. Continuous-Curvature Curve (3C) generation has become one of the key topics for on-going researches in this area. A few types of splines have been proposed to solve this problem. Among them some have been widely used in many applications, which are Bezier curve (B-spline), clothoid curve and cubic spline. Yamamoto et al (1999) studied the B-spline-based path planning for finding the time optimal trajectory; Tomas et al 2003 used the Bezier curve in path planning, having considered minimizing the square of the arc-length derivative of curvature along the curve. Kanayama and Miyake 1986 suggested that using clothoid curve would form a smooth path with continuous curvature, and the resulted curve curvature varies linearly along the path. Later, Y.Kanayama and Hartman (1989) proposed another solution using cubic curve to the same problem. These three basic types of curve were well accepted by many researchers and the research effects made on the applications of using these curves were reported also. Scheuer (1996) and Fraichard (2001) used clothoid curves in their car-like vehicle control experiment. Nagy and Kelly (2001) adopted cubic splines in their trajectory generation algorithm. However, previous works have mainly been focused on the static trajectory generation problem and on finding the solutions for 2D applications.

In this paper we propose a continuous motion trajectory generation algorithm to solve the global trajectory generation problems. CMTG is designed to generate a C^2 smooth 3D curve according an arbitrary path defined by a number of dynamically allocated points in a 3D space for real time applications. This approach can offer a more flexible solution for 3D trajectory applications than others. We have also implemented CMTG into a 3D virtual environment for an aircraft simulation project. This paper is organized as the follows: In the next section, we give a brief review of the three basic types of curve and introduce some of the notations used in this paper. The details of our approach are described in Section 3. Section 4 discusses the simulation examples and results. A conclusion is given in the last section.

CONTINUOUS CURVATURE CURVES
Continuous curvature curves have received more and more attention in recent years for trajectory generation, due to the fact that a continuous curvature curve has a continuous secondary derivative along a path and allows the smooth changing of moving agent angular velocity. Therefore, it satisfies most dynamic systems and eases the trajectory control tasks. Three important types of curve are discussed
in this section, which are Bezier curve, Clothoid curve and Cubic spline.

**Bezier Curves**

The Bezier curve approximation was developed by French engineer Pierre Bezier. Bezier curve has a number of properties that make them highly useful and convenient for curve and surface design. One thing that makes B-spline attractive is the ease by which the shape of the resulting curve can be controlled, the other advantage of B-splines compared to intrinsic splines (such as clothoid curve and cubic spline) is that it can be given in a closed form. However, there are also some disadvantages of this curve, especially when it is used in designing the motion trajectory. Since the parameter of Bezier curve is not time or distance dependent, the curves generated cannot be used directly as a trajectory (Hwang et al. 2003). Another inconvenience is that the curvature of this curve cannot be expressed in a simple profile. A general expression of Bezier curve uses \( n+1 \) control points, denoted as \( p_k = (x_k, y_k, z_k) \), with \( k \) varying from 0 to \( n \). These coordinate points are blended to produce the following position vector \( P(u) \), which describes the Bezier curve between \( p_0 \) and \( p_n \).

\[
P(u) = \sum_{k=0}^{n} p_k \text{BEZ}_{k,n}(u) \quad 0 \leq u \leq 1 \tag{1}
\]

The Bezier blending functions \( \text{BEZ}_{k,n}(u) \) are the Bernstein polynomials

\[
\text{BEZ}_{k,n}(u) = C(n,k)u^k(1-u)^{n-k}
\]

Where parameters \( C(n,k) \) are the binomial coefficients

\[
C(n,k) = \frac{n!}{k!(n-k)!}
\]

**Clothoid Curve**

The clothoid curve is very useful because its curvature varies linearly along the arc. Kanayama (1986) proposed to use this curve for motion trajectory design. It is now the most commonly used curve type for highways and railroads design. It is one of the simplest examples of a curve than can be constructed from its curvature, although the math is somewhat complex. Clothoid curve is intrinsic spline, it can not be expressed in a closed form, and this is the biggest disadvantage and result in calculation difficulty. However, its curvature varies linearly along the arc, and can be constructed from its curvature; on the other hand the parameter \( t \) is proportional to the length of the arc, therefore it can be used directly as a trajectory. The derivative of the curvature of clothoid curve is a constant; this is identical with the Bang-bang control theory, which aims at giving solution to the optimal-control problem. For example, minimize the steering time. The clothoid curve is defined as follows:

\[
Cv(s) = k * s + Cv_0 \tag{4}
\]

Where \( s \) is the arc length, \( Cv(s) \) is the curvature and \( k \) is the coefficient. The direction of the tangent vector is the integration of the curvature, and is expressed by Eq. (5)

\[
\theta(s) = \int_0^s (k \ast s + Cv_0)ds = \frac{1}{2} k \ast s^2 + Cv_0 \ast s + \theta_0 \tag{5}
\]

Furthermore, the coordinates \( x \) and \( y \) are expressed in term of the direction \( \theta \).

\[
x(\theta) = \frac{1}{(2k)^{\frac{3}{2}}} \int_0^\theta \cos \frac{\theta}{2} d\theta \tag{6}
\]

\[
y(\theta) = \frac{1}{(2k)^{\frac{3}{2}}} \int_0^\theta \sin \frac{\theta}{2} d\theta \tag{7}
\]

**Cubic Spline**

Cubic spline is another kind of intrinsic spline for motion trajectory problem (Nagy and Kelly 2001). The curvature of cubic spiral also can be expressed in a simple profile, although it does not linearly vary along the arc. Some authors argue cubic spline is a more natural and smoother set of curves (Nagy and Kelly 2001 and Liang et al. 2002). Smoothness is defined as magnitude of the derivative of curvature. They claimed the set of cubic spiral is theoretically more meaningful. As an intrinsic spline, cubic spline also cannot be given in a closed form. The curvature of the curve includes second power item of the arc length, which will introduces more errors while controlling the motion of a real agent. The cubic spline is defined as follows:

\[
Cv(s) = \frac{6\alpha}{l^3} s(l-s) \tag{8}
\]

Where \( \alpha = \theta(l) - \theta(0) \) is the relative angle of the two separated configurations which have been connected. \( l \) is the length of the spiral.

\[
\theta(s) = \theta_0 + \int_0^s Cv(s)ds \tag{9}
\]

\[
x(s) = x_0 + \int_0^s \cos \theta(s)ds \tag{10}
\]

\[
y(s) = y_0 + \int_0^s \sin \theta(s)ds \tag{11}
\]

A short conclusion about the above three continuous curvature curves can be made now: Bezier curve’s merit is the ease by which the shape of the resulting curve can be controlled and calculated. Its disadvantage is that the curvature of the curve cannot be expressed in a simple form and the parameters of the curve are often not uniformly distributed. The properties of clothoid curve and cubic spline are contradictory and they can not be expressed in a closed form, which, therefore, results in calculation difficulties. However, the curvatures of the curves can be expressed as a simple profile which eases the trajectory control tasks.

**A CONTINUOUS MOTION TRAJECTORY GENERATION ALGORITHM**

In the previous section we have reviewed three types of continuous curvature curves, which are the most widely used curve types in many practical applications. Each of them has its own merits and drawbacks. In this section, we will describe the motion trajectory generation algorithm proposed in our work. As far as we know, most of the
works reported in this area aimed at solving the static trajectory generation problem, and offered the planar solutions, which is only suitable for 2D environments. These algorithms offer single piece (or pair) of smooth curves to achieve a local solution, however, they can not guarantee the smooth transition between individual curves. In complex environments, especially in an environment where uncertain factors may occur, the final trajectory must be made up or blended by a set of individual curves. This blended path must match tangency and curvature at each joints, exhibit continuous curvature and be well behaved with respect to the choice of the control points. Traditional algorithms can not meet these requirements.

Our research endeavours to solve these problems by using a sequence of dynamically allocated points. The clothoid curve is used as the basis for trajectory generations, though other types of continuous curves can be used in a similar manner. Three points in a 3D space are normally used for the curve element configuration. An example is shown in Figure 1(a). S1, S3 and S7 are the points generated according to the environment by any path planning algorithm. Traditional algorithms use symmetric clothoid curve element to connect the start and end configurations. From the point of view in a local area, the agent does follow the curve smoothly from S1 to S7. There is no abrupt change either in the direction of the agent movement or in the derivative of the direction. However, the method above may encounter difficulties in a global configuration. The global trajectory is made up of a set of local curve pieces and we should not only guarantee the smooth agent movement along the local curve, but also guarantee the smooth transition between these curve elements. We propose a novel approach to use un-symmetric clothoid curve element plus two extra dynamic control points S2 and S4 to offer the more flexible and powerful solution to the trajectory generation problem. In the real world, a symmetric pattern cannot always be guaranteed to be the optimal trajectory to follow, and the unsymmetrical pattern is the general situation and offers more flexibility to control process. A simulation of asymmetrical clothoid curve element is shown in Figure 1(b). S1 and S7 are the points selected according to the kinematical states of the moving agent (i.e. speed, direction).

\[
S_i = \lambda S_{i-1} + (1 - \lambda)S_{i+1} \quad 0 < \lambda < 1
\]

\[
\dot{\lambda} = f(u) \quad (u \text{ includes the agent’s speed, direction})
\]

To achieve a successful smooth connection at the joints between curve elements, keeping in mind that each destination state is the new initial state at the same time, the direction of the motion trajectory at the current destination location should be the same as at the next new initial agent location, and the curvatures at the both locations should also be the same for smooth curvature transition. To meet these requirements, two transition points (S2, S4) are introduced which are determined by the motion kinematical states of the agent to produce a smooth transition between curve elements.

![Figure 1](image1.png)  
**Figure 1.** An asymmetrical clothoid curve.

![Figure 2](image2.png)  
**Figure 2.** The agent’s transfer via sequences of curve pairs in a 3D environment

As shown in Figure 2, the agent starts at S1, S1 S2 S6 are the original three points, and S7, S4 are the two points added to control the agent movement to make a smooth transition between adjacent clothoid curve elements. The two control points are derived from the agent motion requirements that satisfy the agent kinematical conditions. The first local curve element ends at the destination point S7, which is also the new initial position of the next clothoid curve element. At S7 the curvature is decreased to zero and the direction is from S7 to S6. The trajectory generation will keep going on as long as subsequent path points supplied. The algorithm can be expressed as follows.

**Step 1:** If New-Process, find the sub-goal (S3)
- Get the Agent and control points’ information
- Decide the position of S2 and S4 according to the agent’s state
<table>
<thead>
<tr>
<th></th>
<th>Trajectory Length (unit)</th>
<th>Total Time cost (ms)</th>
<th>Average end position error (unit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bezier Curve</td>
<td>1131.405</td>
<td>240</td>
<td>0.0</td>
</tr>
<tr>
<td>Clothoid curve</td>
<td>1049.296</td>
<td>280</td>
<td>0.0445</td>
</tr>
<tr>
<td>Cubic Spline</td>
<td>1033.071</td>
<td>280</td>
<td>0.0768</td>
</tr>
</tbody>
</table>

Set the curve parameter.
Else
Move the agent.
End local process while reaching $S_4$, $S_5$.
If sub-goal is the same with the goal, finish;
Otherwise go to step 1.

The clothoid curve element for trajectory generation in our work is implemented in a 3D space via matrix transformation. The equations can be summarized as follows:
\[
\frac{d\theta(s)}{dt} = k \cdot s + C_v
\]
(14)
\[
(\alpha, \beta, \varphi)^T = R(\eta, u) \cdot (\theta, 90 - \theta, 0)^T
\]
(15)
\[
(x\ y\ z)^T = (x_0\ y_0\ z_0)^T + \int_0^s \cos(\alpha\ \beta\ \varphi)^T \ ds
\]
(16)

Eq.14 is expression of the curvature of the basic clothoid curve element. $(\alpha, \beta, \varphi)^T$ is the vector of $ds$ in a 3D space, $(\theta, 90 - \theta, 0)^T$ is the vector of the projection of $ds$ in 2D sub-space. $\eta$ is the angle between $ds$ and its projection. $u$ is the direction of the axis which the projection around. $(x\ y\ z)^T$ is the coordinates of the end position, $(x_0\ y_0\ z_0)^T$ is the coordinates of the initial position.

**SIMULATION RESULTS**

A number of simulations have been conducted in our work for evaluating the motion trajectory generation algorithm. Figures 3 a. b. and c show the results of a comparative experiment of different curves. Three types of curve were used, which are Bezier curve, Clothoid curve and Cubic Spline.

In this experiment, each test used three points $P_1\ P_2\ P_3$ to configure the shape of the curve. From the simulation results, we can see that under the same configuration conditions, Bezier curve has the longest arc length and the biggest absolute curvature value. Therefore, it is sharper than the others. Cubic spline has the shortest arc length and the smallest absolute curvature value. The direction of cubic curve varies more smoothly and the cubic spline is the smoothest among the three candidates. clothoid curve is close to cubic spline, it has small absolute value of curvature and short arc length, the significant merit of clothoid curve is that its curvature profile can be expressed as a linear function of the arc length while the others can not. Clothoid curve can achieve smooth transition with low price ($S$, $C_v$), and it is therefore easier to be used in agent motion control applications.
successful and the results are satisfactory. Figure 6 shows a screen capture of real-time aircraft simulation in a virtual environment, in which the aircraft acted as an autonomous agent who used visual information to detect obstacles in order to find a path to conduct an obstacle-free navigation.

Figure 4 Navigation Simulation Case 1.

Figure 5 Navigation Simulation Case 2.

The algorithm has been implemented in C++ and tested on a 1000 MHz Pentium processor PC with 256M memory and Matrox G450 graphics card (360MHz, 32M memory). The experiment of indoor scene simulation shown in Figure a 5 is using a 500*500 units’ space. There are seven local curve pieces and 5000 points have been produced for the approximation. The time cost of trajectory generation, the accurate trajectory length and the average end position error are shown in Table 1.

Figure 6 A aircraft simulation in a 3D environment

CONCLUSIONS

We have proposed a $C^2$ curve generation algorithm called CMTG for motion trajectory generation problems in a 3D space. The algorithm is designed for real-time applications and to generate smooth trajectory automatically, which follows an arbitrary path defined by a number of dynamically allocated points in a 3D space. Compared with other methods, this approach is more efficient and offers a more flexible solution for real 3D trajectory planning and control applications, either in a simulated environment or in the real world. In addition, it can guarantee the continuous of agent’s motion state. The approach has solved some drawbacks of traditional algorithms and can be used in global optimal applications. Three popular Continuous-Curvature Curves have been studied and tested in our work. The clothoid curve has been chosen as the basis for the motion trajectory generation. A 3D aircraft simulation has been conducted and the result is quite promising. Our future work will be looking into how to develop the approach further for incorporating motion dynamics.

REFERENCES


AUTOMOTIVE BASED SIMULATION
PREDICTION OF THE EFFICIENCY OF AN AUTOMOTIVE OIL STRAINER: COMPARISON OF EXPERIMENTS WITH NUMERICAL SIMULATIONS

Sydney Tekam, Marc Demoulin
Renault S.A
67 rue des Bons raisins
92500 Rueil Malmaison
France

Sydney Tekam, Virginie Daru
ENSAM, SINUMEF laboratory
151 bd de l’Hôpital
75013 PARIS
France

KEYWORDS
Computer Aided Engineering, Aerosol simulation, oil separator, oil droplets, blow by

ABSTRACT

The design and set-up of an oil strainer consist in calculating the pressure drop between the outlet and the inlet of the separator and predicting its efficiency (which is, for each droplet diameter, the ratio of oil droplets trapped in the strainer to oil droplets entering the strainer). A numerical simulation method has been developed at RENAULT, but no experimental correlation had ever been performed. It is done here with a test oil separator designed for this purpose. The efficiency of a test oil separator is evaluated with a measurement bench and a comparison is made with numerical simulation. Thus, accuracy and relevance of the numerical method employed will be quantified.

INTRODUCTION

Blow by gases circulating along the cylinders in an automotive engine are collected and filtered in the oil strainer also called oil separator. These gases are loaded with small oil droplets (average diameter is 1 micron and the diameter range is [0.5μm - 10μm]). The filtration process of blow by gases aims at two different goals: diminish the loss of oil due to blow by circulation; allowing the blow by gases to be re-injected and mixed with fuel in cylinders for combustion, with as less exhaust emission as possible (S. H. Hill, S. J. Sytsma, 1991).

To generate blow by two options were possible: an engine or an aerosol generator. The engine requires an engine-test bench to which the separator should be plugged. The advantage of this solution is to introduce in the separator blow by flow produced by an engine in standard working conditions. Since the engine flow is quite unsteady it is hard to know the exact distribution of oil droplets diameter entering the separator and to determine precisely the efficiency. On the other hand, the use of an aerosol generator should give an unambiguous blow by flow in terms of volume flow, oil mass flow and oil droplets diameter. The generator is able to produce a steady aerosol flow so that the diameter distribution will be of the same order of an engine blow by.

There are many measurement devices for particle sizing as explained by (Kleitz and al.). Two main criteria have driven the choice of the measurement tool: the minimal size of droplets detectable and the measurement sensibility - i.e. the maximum gap between two sizes of particle measurable. In fact, some of the oil droplets diameter goes down to a few tens of microns. The measurement media should also be non-intrusive. A statistical device should the most adapted to such measurement with a high amount of information to analyse (i.e. high quantity of droplets). Therefore, a spectrometer has been used. It covers the right sensibly range [0.5μm - 40μm] and accuracy of the measurement is fine enough, with a separation power of 32 class per decades.

Eventually, a bench is build with a generator, a prototype separator and a spectrometer. Several types of blow by flow are simulated according to of measurements made on a standard F9 engine. Thus, each case stands for an engine working point (load and power) and is defined by a volume flow an oil mass flow and a droplets size distribution. The efficiency of the test-separator is measured for each of the case. Besides, for each case, the configuration of the test separator is changed several times (i.e. either the size or the number of chicanes are changed).

Then, the case performed experimentally has been simulated numerically according to existing numerical methods already developed by (K. Satoh, T. Kawai, M. Ishikawa, T. Matsuoka, 2000): the results are of the same order. Nevertheless, efficiency curves do not have the same shape. Numerical efficiency gives constant values until 5μm whereas it has been found that efficiency rises sharply with size. Additional result will help to make a statement on the reliability of numerical simulations.

EXPERIMENTAL SET-UP

The following sketch itemises the experimental set-up and its components.
Aerosol generator

The aerosol generator is made of a heated oil reservoir with a Laskin nozzle dipped inside (A. Melling, 1997). A pressurised airflow is introduced in the volume of liquid oil through the nozzle. Air bubbles, produced inside liquid oil, escape and absorb small liquid oil droplets. At the reservoir surface the air, bubbles explode liberating oil droplets that are pumped out of the reservoir. An additional airflow is added to get air-oil droplets mixture. The size of the droplets is driven by the type of oil and operating temperature. In fact the oil viscosity triggers the formation of droplets (one can roughly say that the lower the viscosity is, the lower the size of the droplets will be), which depends from oil temperature. The oil mass flow is triggered by the air volume flow passing through the nozzle: the higher is the incoming airflow, the higher will be the quantity of air bubble in oil and thus the higher will be the quantity and concentration of oil droplets leaving the liquid oil volume.

Therefore, the choices of oil, of temperature and of air volume flow in the oil reservoir are the key-parameters controlling the aerosol constitution. For each simulated case, the oil type and the oil temperature are fixed. Only aerosol oil mass concentration will change along the experiment, not the droplet diameter spectra.

Spectrometer

The spectrometer working principle is based on Mie and scattering light theories: a droplet passing through the measurement chamber is lighted; the reflected intensity is measured by a photo-sensor. A calibration curve makes the correspondence between light-intensity measured by the sensor and particle diameter. The spectrometer is equipped with a T-shape measurement cell in order to avoid border zone error (when a particle passes at the border of the measurement cell and reflects less light than it should its size is under-predicted) and coincidence error (when two particles pass in the cell simultaneously and reflect more light than a single one, which leads to a diameter over-prediction). The spectrometer is able to give a particle diameter with an accuracy of 32 sizes per decade. Since 5L/min of the blow by flow must pass through the separator for measurement, down the spectrometer is plugged a suction pump. In order to avoid coincidence error the concentration of measured aerosol must be under $c_{\text{max}} = 110^{5}$ particles/cm$^3$. In such a case, a diluter is added before the spectrometer measurement chamber. In the diluter, the blow by gas flow is expanded by adding a particle free airflow. The drop of pressure enhances a drop of the particle concentration in the aerosol flow, which can be divided by 10 or 100. Since much time was lost in the first cases tested to plug either the outlet or the inlet on the separator, an automated bypass enables a quick switch between the inlet and outlet flow.

Measurement chamber and post-treatment station

The oil strainer is placed in an insulated chamber. It is heated from the bottom by a hot plate. Ducts and other system elements - except the generator - are insulated as well. Therefore the whole system is maintained at the average temperature of 80° C and any condensation of oil vapour fraction carried by the aerosol in low temperature area is prevented.

The bench central control station includes a generator controller, for oil temperature, some manometer, to measure pressure drop between inlet and outlet, flow-meter, and the post processing station of the spectrometer that measures the diameter spectra, particle concentration and efficiency of the separator.

Separator

The separator tested is a prototype chicane-separator designed according to industrial requirement. It is made with 3 length-adjustable and removable chicanes (see figure2). This design requirement requires the lowest pressure drop as possible between inlet and outlet. More over, velocity maximum must not exceed 2m/s in order not to remove the oil film formed by particles deposition. A divergent at the inlet and a convergent at the outlet ensure a progressive raise or decrease of the velocity, between the inlet hole and the first chicane, and between the last chicane and the outlet hole.

Figure 2 Prototype oil separator

Height configurations of the separator are studied - depending of the simulated blow by.
Table 1: Definitions of oil separator configurations

<table>
<thead>
<tr>
<th>CONFIG</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cfg1</td>
<td>3 chicanes - long size</td>
</tr>
<tr>
<td>Cfg2</td>
<td>3 chicanes - medium size</td>
</tr>
<tr>
<td>Cfg3</td>
<td>3 chicanes - small size</td>
</tr>
<tr>
<td>Cfg4</td>
<td>1 chicine - long size - front position</td>
</tr>
<tr>
<td>Cfg5</td>
<td>1 chicine - medium size - front position</td>
</tr>
<tr>
<td>Cfg6</td>
<td>1 chicine - small size - front position</td>
</tr>
<tr>
<td>Cfg7</td>
<td>1 chicine - long size - rear position</td>
</tr>
<tr>
<td>Cfg8</td>
<td>No chicine</td>
</tr>
</tbody>
</table>

Efficiency measurement proceeding

Case measured
The simulated blow by flow are given below (it is important to remember that the spectrometer needs a volume flow of 5 l/min, taken either for the inlet or outlet ducts). Therefore, the efficiency simulation will be performed with an inlet velocity adapted to the added flow measurement.

Table 2: Aerosol constitution for generator settings

<table>
<thead>
<tr>
<th>Qbb (L/min)</th>
<th>Qoil (g/min)</th>
<th>Oil rate per volume (mg/l)</th>
<th>Concentration (mg/m³)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>17</td>
<td>0.2</td>
<td>2.2e-8</td>
</tr>
<tr>
<td>C2</td>
<td>43</td>
<td>1.08</td>
<td>4.7e-7</td>
</tr>
<tr>
<td>C3</td>
<td>21.5</td>
<td>0.15</td>
<td>1.3e-7</td>
</tr>
<tr>
<td>C4</td>
<td>42.5</td>
<td>1.48</td>
<td>6.6e-7</td>
</tr>
<tr>
<td>C5</td>
<td>29.5</td>
<td>0.06</td>
<td>4.1e-8</td>
</tr>
<tr>
<td>C6</td>
<td>100</td>
<td>1.48</td>
<td>6.6e-7</td>
</tr>
<tr>
<td>C7</td>
<td>150</td>
<td>1.48</td>
<td>6.6e-7</td>
</tr>
</tbody>
</table>

Each test performance falls into 3 parts: setting of the spectrometer and of the generator; setting of the separator; data processing and efficiency measurement.

Setting of the generator
The spectrometer setting aims at getting the calibration curve, that gives the size of particle with respect to the light intensity detected by the photo-sensor. The lens of the optical system are cleaned and the suction pump is installed. A mono dispersed dust is introduced. This dust consists of small particles of a fixed diameter, 1 μm in the present case. Therefore, after calibration, the spectrometer should measure a Dirac-shape distribution of the size of the dust. The generator is controlled by the inlet volume flow, the quality of oil and the temperature. Thus, for a given flow, the change of oil or of temperature affects the diameter distributions. The oil has been chosen and the temperature is regulated to have a particle droplet diameter spectra concentrated around 1-2 μm. To get the right oil mass flow the air flow incoming to the nozzle of the generator must be set up. This setting is particularly precise and critical: The oil mass rate is set - i.e., volume flow incoming the generator- and a measurement is made with the spectrometer. When the right value is obtained, according to the spectrometer, a second measurement is made with an absolute filter during 300s and oil mass flow is calculated. The fraction between spectrometer flow and the filter flow is a correlation factor. The volume flow inside the generator nozzle is adjusted according to this correlation coefficient. Another measurement of oil mass flow is made with the absolute filter and the process is repeated until the calibration factor is close to one and the oil mass flow is correct. Eventually, a clean air volume flow is added to get the right aerosol volume flow. With the correct oil mass flow, the desired particle diameter spectra and the right aerosol volume flow, the blow by is fully determined. For highly concentrated blow by flow, a diluter is added to the measurement channel. It is located just before the spectrometer. A factor ten diluter is required for case c2, c4, c6, and c7.

The measurement process consists in two acquisitions of 290s: one where the spectrometer is plugged to the inlet, and then to the outlet. Each measurement is made twice.

Error sources
For each case, a first measurement is on an empty duct. This step estimates the "efficiency" of the measurement system since a simple bend or elbow on a duct can generate separation. The system efficiency is found to be of an average 2% - 3%. The error made on spectrometer measurement is assumed negligible. The error on the oil mass flow calculation with the absolute filter is only generated by the electronic balance used to weight the filter and it is neglected.

Results
The experiments are still in progress and only case c2 and case c3 have been simulated. For each case, eight configurations of the separator have been tested. The separator appears to be poorly efficient (see figure 4). In fact, if the design ensures a very low-pressure drop ΔP~23 Pa, the velocity of aerosol mixture inside the separator is far too low to ensure a good separation of droplets and air. Actually, except config1 of case 2, no other result can be accounted: the average efficiency is between 0 and 3%. This value is similar to the range of error of the system.

Efficiency of the separator for config1 is quite low for values under 3 μm then a drastic rise occurs above 3 μm (from 5% to 3%) This evolution was quite predictable: the bigger the droplets are, the easier they should impact on the chicanes. This experimental case can be compared with a numerical simulation, since efficiency values are not too low.

Numerical simulation
The numerical calculation of a separator efficiency falls into three steps: mesh generation of the separator, flow calculation in the separator, oil particles trajectory calculation.
Mesh generation

Separator mesh is realised, from a numerical sketch of the separator, with software ANSA and TGRID. ANSA is the CAD and Grid generation software. It allows visualisation and generation of both two dimensions and three dimensions items. Its use is quite straightforward, since mesh can be generated from numerical sketch - as CATIA files - of any automotive components. ANSA generates a triangular cell skin mesh of the separator. The mesh is unstructured. The characteristic cell-length is 1mm: the mesh is fine enough and local refinement is not necessary (P. D. Swanson, F. J. Muzzion, A. Annapragada, A. Adkei, 1996). From this skin mesh, T-Grid generates a volume tetrahedral mesh. The separator is quite heavy (V=0.8L) and the mesh size is consequent: over 500000 faces and about 2.5 millions cells.

Flow calculation

Calculations are performed with FLUENT6 code. The fluid is warm air at T=80°C, with a density $\rho=0.96$ kg/m$^3$ and a viscosity $\nu=2.18 \times 10^{-5}$ m$^2$/s$^2$. The incoming blow by flow is 42L/min and the inlet velocity is $V=2.28$m/s, hence an inlet Reynolds Number is $Re=2032$: a K-Epsilon model is used for flow simulations, with a second order resolution scheme. As during the experiment, the flow is steady. The outlet boundary condition assumes $P=1$ bar at the outlet.

The flow in the separator is slower than expected. In fact the flow velocity drops down to 1m/s in the inlet divergent (see figure 3). Then, flow is lightly accelerated after the first and the last chicanes. This acceleration combined with a drastic change of direction should drive the oil particles to hit the wall.

Nevertheless, since average velocity is quite low, the acceleration is quite limited. The pressure drop which is very low $-DP=5.19$ Pa- is consistent with the low velocity magnitude variations along the flow path. Some recirculation should occur after the chicanes. The recirculation area participates in the separation process since oil particles trapped in circulation stay longer in the separator and thus have more chance to be trapped. On the other hand, recirculation also participates in the loss of pressure between inlet and outlet. Velocities in the separator are too small and no recirculation occurs (see figure 3).

![Figure 3: Velocity vector colored by velocity magnitude in the inlet of the separator](image)

Oil droplets trajectory

Oil droplets are small and spherical. They are assumed to be solid spherical particles of constant diameter. These particles are introduced in the separator. Each particle motion equation is given below (M. R. Maxey, James J. Riley, 1983):

$$\frac{d u_p}{dt} = F_D (u - u_p) + P$$  \hspace{1cm} (1)

$F_D$ is the particle drag coefficient given by Morsi and Alexander, 1972

$$F_D = \frac{18 \mu C \frac{D_p}{D_p^2}}{\rho_p D_p^2 12} \hspace{1cm} (2)$$

With P standing for other inertia forces, $u$ as fluid velocity, $u_p$ as particle velocity, $\rho$ as air density, $D_p$ as particle diameter and $Re_p$ particle mass Reynolds number. According to Morsi and Alexander, for spherical particles:

$$C_D = a_1 + \frac{a_2}{Re_p} + \frac{a_3}{Re_p^2} \hspace{1cm} (3)$$

FLUENT integrates each trajectory with a variable time step. Time step is adjusted so that a particle path crosses one cell per iteration.

Particle inlet velocity is assumed equal to the air velocity. It is assumed that in simple duct, the oil droplets are carried by the airflow without any inertia effect. The particle-trapping model is simple: when a particle path hits a wall, the particle is marked trapped and the trajectory stops. If the trajectory hits the surface of the outlet, the particle has escaped the separator. The efficiency is the ratio of the number of particle trapped above the number of particles introduced.

An injection of 2000 particles is made for 10 droplets diameter [0.3; 0.5; 0.7; 1; 1.2; 2; 2.5; 3; 3.5; 4; 5] (size given in µm). Thus, the curve of efficiency vs. particle diameter is drown.
Eventually, the numerical calculation predicts a constant and very low efficiency: 5%. This result is consistent with flow calculation: a low velocity inside the separator means that changes of direction are not fast enough. Thus, particle paths can follow streamlines from inlet to outlet and avoid the wall. Nevertheless, one should expect the efficiency to rise with the droplets size.

**Numerical efficiency - vs.- measured efficiency**

Both experimental and numerical simulations have predicted a poor separation power. The separator generates a low-pressure drop and a low velocity for air. The experiment predicts a raise of the separation power above 3μm whereas numerical results are constant. Since the numerical procedure is simple and straightforward, some improvements have been proposed to upgrade the method. Impact treatment and particles trajectory integration must be improved to get efficiency closer from the experimental values.

Besides, the experiment values are too low to be reliable. Experiments are still in progress and additional test will be performed: first, with a higher aerosol volume flow (as for cases c6 and c7) at the inlet in order to have higher velocity in the separator and ensure a better trapping; then with a cyclone separator which is more efficient with small particles (W. Krause, K. H. Spies, L.E. Bell, F. Ebert, 1995). These two methods aim at getting higher efficiency curves so that error can be minimised in comparison with effective efficiency.

**CONCLUSION**

A method for simulating blow by flow and oil separation in an engine oil separator is presented. Both experiment and numerical methods have been applied on a prototype separator. The results are different: the efficiency spectrum is constant for numerical simulation but increasing for experiment. The separator was proved to have a very low separation power. This first series of results is insufficient to make a statement but experiments are still in progress in order to get reliable efficiency values. Besides, numerical method upgrading should shift up calculation accuracy. New experimental and numerical result will be presented at the ISC2004 congress.

**REFERENCES**


A. Kleitz, D. Bouloud, "Granulométrie des particules en mouvement et des aérosols", *Technique de l'ingénieur*, R 2 360

W. Krause, K. H. Spies, L.E. Bell, F. Ebert, 1995,"Oil separation in crankcase – New concepts through system analysis and measurements", *SAE technical papers series*, 950939


**BIOGRAPHY**

Sydney TEKAM was born in Clamart in 1978. During his graduation of Engineering in Hydraulics and Fluid Mechanics at National Institute of Engineering in Electro technology, Electronics, Computer Science, Hydraulics and Telecommunication (Toulouse, France) he has worked on the simulation of oil vapour concentration in the ring pack crevice, in collaboration with the Sloan Automotive Laboratory (MIT, Cambridge, USA). He has attended the Master of Science of University of Manchester Technology Institute in Thermal Power and Fluids Engineering. Then he has participated, on behalf of EdF-Chatou (Electricité de France Research Department, Chatou, France) to the implementation of a linear-production model to a k-epsilon based CFD code. This model was used for the simulation of the flow around an Ahmed body. This work was presented at the 2001 ERCOF'TAC workshop on turbulence. Since March 2002, he is preparing a PhD, sponsored by RENAULT SA, on the simulation of blow-by flow in an oil strainer.
Development of the ECU for the RSBW using a HILS system

Woo-Je Sung, Seung-Kyu Oh, Tong-Jin Park
Dept. of Precision mechanical Eng.,
Hanyang Univ.,
South KOREA
E-mail: swj0000@hanyang.ac.kr
Chang-Soo Han
Professor, Div. of Mechanical Eng., Hanyang Univ., South KOREA
Sang-Ho Lee, Hee-Young Cho, Un-Koo Lee
Hyundai Motor Co., South KOREA

KEYWORDS
ECU (Electronic Control Unit), SBW (Steer by Wire), HILS (Hardware-In-the-Loop-Simulation), RSBW ( Rack-actuating SBW).

ABSTRACT
The SBW system in a passenger car substitutes the typical mechanical linkage between the steering wheel and the front wheel with electric wires. The control conception of the RSBW system was proposed in this paper. Basically, the controller should compensate for a conventional steering system composed of mechanical linkages. The HILS system was developed for the ECU to control an SBW system. The HILS system is available for developing control logic before installing the ECU in a real vehicle. The prototype ECU has confirmed its suitability before applying the SBW system to the real vehicle through a number of simulations with the HILS system.

INTRODUCTION
The SBW system used to enhance vehicle steering has been undergoing worldwide development as the SBW system has many advantages. The SBW system can save energy, reduce weight through elimination of the steering wheel column, and reduce noise and vibration through elimination of a hydraulic pump and gear box [1]. In addition, the SBW system may have space and packaging benefits in the engine compartment, resulting from elimination of the steering wheel column, and the intermediate shaft. In order to develop a system with these merits, research for the SBW system has proceeded rapidly. Sanket Amberkar et al. proposed a system-safety process in a by-wire system for new automotive technology [2]. The rack actuating type has the advantage of satisfying rack system of a conventional vehicle composed of 6-bar mechanical linkage and one motor. In this paper, the RSBW, as an abbreviation, means the Steer-by-Wire system composed of the front wheel steering with a rack actuating type.

In the design of a controller for the RSBW, two factors are emphasized in this paper. The first factor is that the steering wheel motor should generate feeling of steering similar to that in a conventional vehicle. A number of studies have been conducted using the steering assist motor to improve steering feel. Masahiko et al. proposed the use of a controller in an Electric Power Steering system for improving the steering feel [3]. As the steering feel generated by a steering motor is stiff and has low On-center handling performance in comparison to a hydraulic power steering [4], the steering feel in a RSBW system is also investigated in this paper.

The second factor that in emphasized in this paper is that the front wheel motor can improve maneuverability and stability of the vehicle. The RSBW system controller is to guarantee of the mechanical stability, when mechanical linkage of a steering system changes electronic linkage. One of the main aims of research on control of the front wheel linkage mechanism places emphasis on vehicle stability while replacing mechanical linkage with wire mechanisms. Hideki et al. investigated the Electronic Control Unit (ECU) robustness for satisfying vehicle stability [5]. Ryouhei et al. also proposed the vehicle stability control responsibility improvement using SBW [6]. Masaya et al. investigated a vehicle stability control strategy for the SBW system [7].

In this paper, the purpose of the RSBW is for the steering wheel motor control to improve driver’s steering feel. The front wheel motor controller is also proposed to improve vehicle maneuverability and stability based on the mechanism modeling. Consequently, the prototype ECU is developed based on these purposes.

To verify the ECU in a real vehicle, the control algorithm based on vehicle dynamics must be verified and a number of simulations according to the various steering input should be performed. Therefore, the motor properties and various steering performances can be tested and a test environment is required to understand the vehicle stability and dynamic properties according to the steering input. For this purpose, the portion of the real system is replaced by an actuator and is controlled in real time so called Hardware-In-the-Loop-Simulation (HILS) system [10]. Peter et al. developed the controller using the HILS system [11]. The HILS method is applied to test the system control algorithm and the steering system performance in this paper.

This paper is organized as follows: in section 2, the RSBW system modeling is described using bond graph modeling. In section 3, control strategies are introduced to improve vehicle stability, in section 4, the hardware design in HILS system and the ECU design are described, in section 5, verifications with full-car modeling and simulations in HILS system are described for checking the ECU performance. The conclusions of this paper are summarized in section 6.
**RSBW MODELING**

The RSBW system is modeled mathematically using the bond graph method to design the controller [8]. Fig. 1 shows the RSBW system.

![Bond Graph Modeling of RSBW System](image)

**Fig. 1. Rack-actuating steer by wire system**

The RSBW system is composed of the steering wheel motor system generating steering reaction torque and the front wheel motor system generating vehicle steering torque.

The steering wheel motor system is composed of a steering wheel and a motor. The sensors in the steering wheel system are an encoder and a steering torque sensor. The front wheel system is composed of a 6-bar linkage, an AC brushless motor and an encoder for measuring the rack displacement. Fig. 2 shows bond graph modeling of the RSBW system.

Where $K_{mg}$ and $K_{m}$ are motor constants related to motor current and motor torque. The main concept of the steering wheel system modeling is that driver’s steering torque is measured by a torque sensor. Namely, the motor in the steering wheel system does not operate if the steering torque sensor output is zero. The main concept of the front wheel system modeling is that the rack and the motor can be regarded as one body because the ball screw system functions without a mechanical loss between the rack and the motor. Additionally, tire lateral stiffness is regarded as the stiffness of the front wheel system. The inertia of the motor shaft is ignored because it is much smaller than that of the steering column. Equation (1) shows the modeling of the steering wheel system.

$$J_{sw} \dot{\theta}_{sw} + B_{sw} \dot{\theta}_{sw} + K_{mg} \theta_{sw} = T_b$$

Equation (2) is the relation between a driver’s torque and a torque sensor in the steering wheel system.

$$T_d = K_s \cdot T_b$$

$K_s$ in (2) is the tuning variable for modifying a driver’s steering feeling. The main modeling element of the steering wheel system in the RSBW is that the relationship between the driver’s steering feel and torque sensor is linear. Linear modeling of a driver’s torque facilitates tuning of the steering feel. Equation (3) shows the modeling of the front wheel system.

$$m_s \ddot{y}_s + b_s \dot{y}_s + k_s y_s = U_{sw}$$

Where $U_{sw}$ is the force controlled by the front wheel motor. The stiffness of the front wheel system is replaced to the tire lateral stiffness because the tire lateral stiffness is governing variable on the dynamics than that of the front wheel system. The linear relationship between the rack distance and steering wheel angle is modeled by (4)

$$y_s = K_s \cdot \theta_{sw}$$

$K_s$ is the gear ratio between steering wheel angle, $\theta_{sw}$ and rack displacement, $y_s$ in a conventional steering system with mechanical linkage. The control aim of the front wheel motor should satisfy the constraint (4) for following a conventional vehicle steering.

**CONTROL ALGORITHM**

The ECU of the RSBW controls the steering wheel motor and the front wheel motor. The steering wheel motor is a brushed DC motor which has a rated torque 4.07 kgf.cm for 1026 rpm. The front wheel motor is a brushless AC motor which has a maximum torque 9.46 Nm for 450 rpm. The aim
of the controller design is that the RSBW follows a conventional steering system. In this section, the design procedure of the controller is described.

**Steering wheel motor control algorithm**

A steering angle sensor and a torque sensor are located in the steering wheel. A steering angle sensor is used for the front wheel motor control. A torque sensor is used for controlling a steering wheel motor to generate steering reaction torque. The basic purpose of the steering wheel motor control is to generate reaction torque the same as in a conventional power steering system. The stiffness in the steering wheel system is a control object due to its relationship with the driver’s steering feel. If the stiffness in the steering wheel system is high or low, a driver’s steering feel is correspondly stiff or not. As the steering wheel characteristic which returns it to center is determined by the steering system stiffness, the On-center handling performance can be improved by controlling the steering wheel motor. In this paper, the aim of the steering wheel motor controller is to follow the steering torque in a conventional steering system.

![Control signal block diagram](image)

(a) Control signal block diagram

Given:

$$ G_c = \frac{u_{so}(s)}{r(s)} = \frac{14(s + 33)}{s} \quad (5) $$

The control input equation using a deference equation is as follows;

$$ u_{so}(k) = -12.685u_{so}(k - 1) - 4206.75u_{so}(k - 2) + 4219.4u_{so}(k - 3) + 176.37r(k) - 6.49r(k - 1) \quad (6) $$

Where the sampling time is 0.01 seconds. Fig. 3 (b) shows the gain selection for the PI controller. Consequently, the controller of the steering wheel motor is the stiffness controller generating a steering reaction torque for a driver’s torque.

**Front wheel motor control algorithm**

A position sensor measuring the rack displacement is located in the front wheel system. The rack displacement is applied as a feedback signal to the front wheel motor controller. The basic purpose of the front wheel motor control is to compensate the steering angle for driver’s input. The stability of the front wheel motor controller is related to the vehicle stability. If any system is strictly positive real (SPR), the system is stable and has robustness for any disturbance. To be a SPR system, the input-output transfer function of the front wheel system should have the following characteristics: the eigen values of a system are positive definite and the input-output transfer function has one or zero relative degree. The relative degree of an input-output transfer function is two from the front wheel system modeling. (3). The controller of the front wheel motor should be minus one relative degree, that is, additional zero need. Consequently, the Proportional Derivative (PD) controller differentiating the input signal is considered in this paper. As the input signal from the steering wheel system is low frequency, the input and feedback signal can be differentiate reasonably and the ECU with a 16-bit processor can be an adequate differentiator. Fig. 4 shows the controller design for the discrete model of (3). The control signal block diagram in Fig. 4 (a) shows that the front wheel motor is controlled according to reference input, the steering wheel angle generated by the steering wheel system.
Equation (7) is the control input for the continuous system of the front wheel motor.

\[ G_c = \frac{u_{in}(s)}{r(s)} = \frac{1700000(s + 712)}{s} \quad (7) \]

The control input equation using a deference equation is as follows;

\[ u_{in}(k) = -u(k-1)_{in} + 871100000 \cdot r(k-1) + 155000000 \cdot r(k) \quad (8) \]

Where the sampling time is 0.01 seconds, Fig. 4 (b) shows the gain selection for the PD controller. Consequently, the controller of the front wheel motor is the robust controller satisfying the SPR characteristics.

**HARDWARE DESIGN**

In the HILS system, the full vehicle model analysis is processed in the main computer and the steering system is replaced by the HILS mechanism. The ECU executes real time control in the HILS system. This section described the hardware design in the HILS system and the prototype ECU design.

**HILS system**

To construct a HILS system, hydraulic control and input output signal data processing are conducted. It is combined with the vehicle model and the environment for developing an improved RSBW controller is accomplished by real time control. The HILS system consists of an RSBW system, an ECU, a hydraulic system, and a real time controller. The hydraulic system consists of a hydraulic cylinder, a hydraulic controller, and a hydraulic motor. This hydraulic system generates lateral force occurring on the tire when the real vehicle drives with steering action. In addition to the driver’s steering feel efficiency according to the vehicle velocity, the hydraulic system mechanism connected with the RSBW system is applied with the same mechanism transferring the steering force from driver to the tire of the real vehicle. A Digital Signal Processor (DSP) board is used for the real time controller. The mathematical vehicle model is analyzed and the control signal of the hydraulic cylinder is controlled by the DSP results at the same time in the host computer. The load cell signal used for the feedback signal is transferred to the host computer through the DSP board and the state variables showing the overall vehicle movement are outputted. The DSP board used in this paper is TMS320C31 of TI. The real time control process downloads the results of the mathematical vehicle model analyzed from a host computer to the DSP board and transmits the input and output signals from the DSP board to the host computer again for analyzing the mathematical vehicle model. Fig. 5 shows the schematic diagram of the HILS system.

**Fig. 5 Schematic diagram of the HILS system**

The hydraulic system used in this paper plays a role in creating the tire lateral force occurring when the vehicle is doing steering action. The Pulse Width Modulation (PWM)
method is used to control the current because the hydraulic system is controlling the flux by current. A sliding mode control is used for the hydraulic system control. The vehicle velocity is determined first and the tire lateral force by inputting the steering wheel angle is determined second. These values are used to control the flux and move the cylinder after translating to PWM signal. The Pacejka tire model is used most commonly for the mathematical vehicle model [9]. The filter circuit is used for the applied hydraulic system to remove the noise from the signal output and a circuit for digital control is also used. A load cell is used to measure the force at the hydraulic cylinder. It measures the force acting on the cylinder and then the force is inputted into the DSP board as an analog signal after feedback. This value modifies the error to create exact tire lateral force in the hydraulic cylinder according to the steering angle inputs. The former modeled vehicle model is programmed with matching real time control in order to operate HILS. The real time control program is designed on the basis of the mathematical model simulation. The saved data is processed in real time by downloading the program to the DSP board. The real time control model is designed to avoid applying the differential variable or complicated nonlinear function in order to analyze the real time control quickly.

Fig. 6 shows the hardware part and software part in HILS system set up in this paper. Hardware is composed of the hydraulic system and the RSBW system. Software is displayed on screen to see the visualized performance of the RSBW controller in real time.

**ECU design**

The ECU referred to in this paper controls both the steering wheel and front wheel motors at the same time. Fig. 7 shows the ECU block diagram for RSBW system and the prototype ECU for the RSBW system. A 68HC912 by Motorola was used for the micro controller because it is usually used as a main part in a vehicle electronic controller. As shown in Fig. 7, the input signals from a velocity and rack bar displacement sensor are processed as an analog signal.

![ECU block diagram](image)

(a) ECU block diagram

![Prototype ECU](image)

(b) Prototype ECU

Fig. 7 ECU block diagram and prototype ECU

However, the input signals from a steering angle and engine signal are processed as a pulse accumulator of the 68HC912. Based on the input signals, PWM signals are outputted to the timer port of the 68HC912. These PWM signals are transmitted to the motor driver circuit and control the current of each motor. An H-bridge circuit using MOS-FET was used for the motor driver.
These parts are also exclusively used in vehicles at real auto companies. In the RSBW system, a DC motor was used for the steering wheel and an EPS motor was used for the front wheel. The front wheel motor can output up to 30A maximum. A fault detection circuit is added to enable failsafe logic to be applied in the ECU by checking service current of each motor driver.

The yaw-rate, lateral acceleration, and vehicle velocity obtained from full vehicle modeling are inputted in the ECU after they are changed to analog signals in the DSP board. Fig. 8 shows the signal relation in the HILS system. In Fig. 8, the signals of the yaw-rate and lateral acceleration sensor were assumed to be inputted as an analog signal and established to be used with the signal of the HILS output. As the prototype ECU is used the HILS system, the RSBW system control algorithm of the ECU could be verified to see how it affects the full vehicle.

**SIMULATION IN HILS SYSTEM**

In this section, the RSBW system modeling and controller are simulated in HILS system. A full car model is used to HILS system for checking vehicle stability, maneuverability and proper steering reaction force.

**Full-car model**

A full-car modeling including state variables for estimating vehicle stability is needed to examine the vehicle’s driving properties according to desired control and steering input. Fig. 9 shows the full-vehicle dynamics model used in this paper. In the full-vehicle dynamics model, the important state variables are yaw, roll, and the x and y-axis direction motion of the vehicle. Table 1 shows the parameters and each D.O.F. used in the full-vehicle-model.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>VALUE</th>
<th>UNIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle (unsprung) mass</td>
<td>Vehicle: 1460</td>
<td>Kg</td>
</tr>
<tr>
<td></td>
<td>unsprung: 48.25</td>
<td></td>
</tr>
<tr>
<td>Dist. From C.G. to front</td>
<td>2630</td>
<td>m</td>
</tr>
<tr>
<td>Dist. From C.G. to rear</td>
<td>1.402</td>
<td></td>
</tr>
<tr>
<td>Steering wheel inertia</td>
<td>1.298</td>
<td></td>
</tr>
<tr>
<td>Steering wheel stiffness</td>
<td>0.0791</td>
<td>N/rad</td>
</tr>
<tr>
<td>Steering wheel damping</td>
<td>0.00024</td>
<td></td>
</tr>
<tr>
<td>Front wheel inertia</td>
<td>337</td>
<td></td>
</tr>
<tr>
<td>Front wheel stiffness</td>
<td>3.850×10⁻⁷</td>
<td></td>
</tr>
<tr>
<td>Tire lateral stiffness</td>
<td>0.5000</td>
<td>N/m</td>
</tr>
<tr>
<td>Tie-rod length</td>
<td>0.3</td>
<td>m</td>
</tr>
<tr>
<td>Knuckle length</td>
<td>0.3</td>
<td>m</td>
</tr>
<tr>
<td>DC motor constant</td>
<td>0.0558</td>
<td></td>
</tr>
<tr>
<td>BLAC motor constant</td>
<td>0.0515</td>
<td></td>
</tr>
</tbody>
</table>
In Table 1, pitch direction motion is ignored not because it relates to the vehicle steering performance directly, but because it relates to the acceleration and deceleration performance of the vehicle. Table 2 shows the full vehicle parameters used in the full vehicle model. Based on these two tables, the established model could be used to paper the RSBW system. Fig. 10 shows the validation of the full-car vehicle modeling. As shown in Fig. 10, the full-car modeling is employed for checking the performance of the ECU in the HILS system.

![Graph 1: Yaw rate validation](image1)

(a) Yaw rate validation

![Graph 2: Lateral acceleration validation](image2)

(b) Lateral acceleration validation

Fig. 10. The validation of yaw velocity comparison result about a step steering input in 80Km/h

**HILS validation**

The purpose of the HILS validation is to verify if the vehicle state can validate the performance of the RSBW controller as the hydraulic actuator is operating with a mathematical vehicle model for real time control. For this reason, the analysis results of the mathematical full vehicle modeling added an RSBW system to a normal computer. Also, the results obtained from the HILS system through real time control are compared and then the reasonableness was verified. To minimize the output error of the HILS system at low and high speeds, the vehicle state parameters are normalized by maximizing vehicle state variables obtained from the vehicle data. The same steering angles of a computer and HILS system were given and the vehicle’s lateral force, lateral acceleration and yaw rate were outputted and compared. Fig. 11 shows the comparison of the HILS output and the computer simulation output with respect to same step steering input when vehicle velocity is 80km/h.

![Graph 3: Comparison](image3)

(a) When a vehicle’s speed is 40Km/h

![Graph 4: Comparison](image4)

(b) When a vehicle’s speed is 80Km/h

Fig. 11. Comparison of the HILS output and the computer simulation output

As shown in the simulation results, a little error occurred because of the established hydraulic control logic and mechanical properties of the hydraulic cylinder. Nevertheless, full vehicle state output is following the mathematical vehicle model state output simulated in a computer when the vehicle velocity increases. The reason is that the lateral force localized by vehicle weight and stiffness of the front tires is less than the lateral force of a full vehicle. This difference occurs because the dynamic quality of motion increases according to vehicle velocity. That is, the yaw rate and lateral force of front tires are affected minimally by full vehicle lateral force at high speeds.

**ECU output in HILS system**

The performance of the ECU for the RSBW system controller was tested through the HILS system. The performance of the ECU output data was tested by measuring the steering wheel and front wheel torque and then verified by full vehicle state simulation obtained from ECU control. Fig. 12 shows, the comparison of the driver’s steering feel with an RSBW system and a real commercial power steering system when parking.
Fig. 12. Steering feel of the RSBW in parking

As shown in Fig. 12, the steering wheel motor control algorithm following conventional power steering drive’s feel was verified in the HILS system. The driver’s steering feel can be also improved more by tuning the gain parameter controlling the motor. Fig. 13 shows the front wheel motor torque with respect to the steering angle when vehicle velocity is 20km/h and 100km/h.

Fig. 13. Front wheel motor control output

As shown in Fig. 13, front wheel motor torque decreases as the vehicle velocity increases. Using these results, vehicle stability can be improved by the prototype ECU. Fig. 14 shows the simulation results of yaw rate and steering torque variation affecting the driver’s steering feel.

Fig. 14. Steering feel simulation results

In Fig. 14 (a), the steering response is delayed at high speeds because the vehicle has more understeer propensity at high speeds than low speeds. As shown in Fig. 14 (b), the vehicle can drive with more improved vehicle stability at high speeds because of the RSBW system output playing the same steering wheel angle. That is, the driver should put more steering torque input to make the same steering angle at high speeds. Using these results, the performance of the ECU controlling the RSBW system could be tested by the HILS system and the performance can be improved by tuning the gain parameters.

CONCLUSION

The scheme of modeling RSBW system and controlling is described in this paper. The RSBW system is divided into two parts the steering wheel system and the front wheel system. The control algorithm in each part is realized. The RSBW system is defined through changing from a mechanical system to an electronic system. Many types of SBW systems have been suggested. In this paper, the RSBW system is considered. The development of the RSBW ECU using the HILS system was also researched. The steering wheel motor control to improve driver’s steering feel and front wheel motor control to improve vehicle stability was studied.

The conclusions of this paper are as follows:

1. The bond graph method was used to model RSBW system. Modeling was divided into the steering wheel system bond graph and the front wheel system bond graph.

2. In steering wheel system modeling, the relationship between the steering wheel angle and driver torque and application of reaction force are considered for improving steering feel.

3. In front wheel system modeling, describing steering wheel angle to rack bar distance and tire steering angle are control objective parameters.
4. The RSBW system was simulated using the HILS system. By using this system, various types steering is inputted safely in the laboratory and the performance of the RSBW controller could be verified.

REFERENCES


SIMULATION OF A STATE OBSERVER BASED CONTROL OF A VEHICLE SEAT

Dr Charles Stammers
Department of Mechanical Engineering
University of Bath UK
(email enscws@bath.ac.uk)

Dr Tudor Sireteanu
Institute of Solid Mechanics
Bucharest
Romania

KEYWORDS
Semi-active control, state observer, human body.

NOTATION
m mass
u seat relative displacement
x displacement
y road surface input
B damper constant
F control force
K spring stiffness
S_x surface spectral density
Subscripts
a lower internal body
b upper internal body
se seat
s sprung mass

ABSTRACT
A vehicle having a semi-actively controlled seat and sprung mass is studied. The driver is modelled as a double mass system, and a state observer is developed to predict the visceral (internal) displacement of the driver’s body.

The objective is control the seat to minimise visceral acceleration. The logic is able to reduce rms visceral acceleration by forty percent relative to the passive seat case. This is about sixty per cent of the reduction achieved by an idealised fully observable system.

1. INTRODUCTION
The authors have carried out experimental work in the field of semi-active control of vehicles (Stammers and Sireteanu 1998, Guglielmino et al 2004). While it is possible by suitable algorithms to control the sprung mass acceleration, the actual comfort of vehicle occupants is related to internal (visceral) displacements which cannot be measured. There are two lines of investigation, the achievement of an ‘in car’ control system and the development of simulations in which an observer is used to deduce visceral motion.

From shake test measurements of apparent mass of sixty subjects, Wei and Griffin (1998) deduced what they called a one degree of freedom model for a seated occupant, although they identified a second mass which could be the skeleton. This first model indicated an internal body resonance close to 5 Hz. A two degree of freedom model, which contains two internal body masses, was also obtained. An extra resonance around 9.5 Hz was indicated.

The authors (Sireteanu et al 2002) used the latter model in a study of seat control to protect the driver when the vehicle was travelling on rough tracks. It was assumed that the system was fully observable, which of course would not be the case in reality. The response in the higher mode was negligible due to the low level of surface input at 9.5 Hz for realistic vehicle speeds.

2. THEORETICAL MODEL
2.1 Model characteristics
The model assumed here is shown in Figure 1. The significant surface input frequencies at realistic vehicle speeds are below 10Hz. Hence following Cebon (1999) the wheel freedom is not modelled; the surface is assume to act directly on the damper and an effective suspension spring.

Figure 1: Vehicle-Driver Model
When a semi-active element \( F_s \) is assumed in the suspension in order to control the sprung mass resonance, the passive damper \( B_s \) is not present. Similarly, the seat is assumed to have a spring and damper in the passive case. When a semi-active unit \( F_{se} \) is included, the damper \( B_{se} \) is not present.

The unit above consists of Wei and Griffin’s 2 DOF occupant model with masses \( m_a \) and \( m_b \). The mass \( m_{se} \) consists of the third mass (possibly skeletal) derived by Wei and Griffin together with the seat mass. There is no significant cushion flexibility assumed as seat as the control is supposed to remove the need for this.

### 2.2 Equations of Motion

Considering a single internal mass \( m_a \), the equations governing the motion are

\[
\begin{align*}
\ddot{x}_a &= -K_a (x_{se} - x_a) + B_a (\dot{x}_{se} - \dot{x}_a) \\
\ddot{x}_{se} &= -K_a (x_{se} - x_a) - B_a (\dot{x}_{se} - \dot{x}_a) + K_{se} (x_{sp} - x_{se}) + B_{se} (\dot{x}_{sp} - \dot{x}_{se}) + F_{se} \\
\ddot{x}_{sp} &= -m_{se} \ddot{x}_{se} + m_{se} \ddot{x}_a + K_a (y - x_{sp}) + B_a (y - \dot{x}_{sp}) + F_s
\end{align*}
\]

The mass \( m_{se} \) consists of the seat mass together with the skeletal mass which is assumed to have the same motion as the seat. The internal displacement \( u = (x_{se} - x_a) \) is needed to evaluate discomfort but cannot be measured. It is therefore necessary to contract a state observer. The concept was first proposed by Luenberger and followed by Cumming and Newman. However, the proposed schemes suffer various complications (Crossley and Porter 1979). The method outlined here is quite simple.

### 2.3 State Observer

Dividing equation (1) by \( m_a \) and equation (2) by \( m_{se} \) and subtracting

\[
\ddot{u} = -K_a r \ u - B_a r \ \dot{u} - K_{se} (x_{sp} - x_{se})/m_{se} - B_{se} (\dot{x}_{sp} - \dot{x}_{se})/m_{se} + F_{se}
\]

where \( r = (1/m_a + 1/m_{se}) \).

If \( F_{se} \) can be chosen to cancel the term \( K_{se} (x_{sp} - x_{se}) + B_{se} (\dot{x}_{sp} - \dot{x}_{se}) \), \( u \) will decay toward zero (so long as control is possible). The relative displacement and velocity of the seat with respect to the sprung mass has to be known. In practice this might be achieved by a pull-wire transducer.

For the semi-active device to be ‘on’ it is necessary that

\[
F_{se} (\dot{x}_{sp} - \dot{x}_{se}) > 0
\]

If the system were active and the time constant of the device low, the relative displacement and velocity of the visceral could be driven to zero and no discomfort would be experienced. While the magnetorheological damper has a low time constant (less than 25 ms), experience with a semi-active control shows that it can only be ‘on’ about half of the time.

The vertical acceleration \( \ddot{x}_{se} \) of the seat can be recorded by means of an accelerometer (ideally by three to exclude signals due to pitch and roll of the seat). Successive integration of the vertical acceleration yields \( v_{se} = \dot{x}_{se}/dt \) and \( x_{se} \).

The input from the sprung mass can be deduced by means of a pull-wire transducer which records the relative displacement \( x_{se} - x_{sp} \). This has been possible in the laboratory, but whether the ‘in car’ life of such a device is acceptable is unclear.

If the seat suspension forces (spring and damper) are known, \( F_{se} \) can be chosen to cancel this force, thus isolating the seat from the sprung mass input. With a semi-active system it is required that the force \( F_{se} \) should oppose the relative velocity \( x_{se} - x_{sp} \).

If this is violated, the MR damper has to be turned off.

### 2.4 Surface Input

The driver response was computed for a range of sinusoidal inputs defined by the assumed spectral density \( S_g \) of the road profile. This is assumed to be of the form

\[
S_g (\Omega) = C \Omega^{-N}
\]

where \( \Omega \) is the spatial frequency in cycles/m and \( S_g \) has the form \( m^2/(cycle/m) \).

Wong (1978) quotes

\[
\begin{align*}
N &= 2.1, \quad C = 4.8 \times 10^{-6} \quad \text{for a smooth highway} \\
N &= 2.1, \quad C = 4.4 \times 10^{-6} \quad \text{for a gravel road}
\end{align*}
\]

As the exponent \( N \) is the same for both cases there is a simple scaling of the spectral density.

In practice, the ground input is described in terms of a finite number \( n_w \) of sinusoidal waves, the root mean square amplitude of each being \( [S_g(\Omega) \Delta \Omega]^{0.5} \) where \( \Omega \) is the mid-band spatial frequency and \( \Delta \Omega \) is the bandwidth chosen.

\( \Delta \Omega \) can be variable but here is chosen to be the same for all bands. If the vehicle speed is \( v \), the ground input frequency \( f \) (Hz) is \( \Omega v \).

Random phase in the range \([0, 2 \pi]\) can be assigned to each wave and the overall ground input obtained as a single multi-frequency signal. If so, in order to avoid ‘beating’
effects, it is advisable to ensure that the spatial frequencies chosen are not integer multiples. A simple stratagem is to choose centre band values of the form $\Omega + \pi / k, 2\Omega + \pi / k, 3\Omega + \pi / k$, where $k$ is some integer.

2.5 Frequency effects

Discomfort is governed by the human body response. It is known that discomfort is frequency dependant, and hence it is advisable to apply weightings to the response. In practice this calls for single frequency inputs. This is not possible with the complete multi-frequency input unless an FRA is carried out, although if the system is linear the actual inputs can be weighted. However, it is known that that internal body natural frequencies vary with input vibration amplitude.

The weighting $w_{tg}$ used for vertical vibration of a seated occupant for $f$ in Hz is

$$w_{tg} = \begin{cases} 0.4, & f < 2 \text{ Hz} \\ f/5, & 2 \text{ Hz} < f < 5 \text{ Hz} \\ 1.0, & 5 \text{ Hz} < f < 16 \text{ Hz} \end{cases}$$

A measure of the total weighted acceleration $A_{\text{TOT}}$ of the occupant is taken to be

$$A_{\text{TOT}} = \sum [w_{tg}(j)a(j)]^2$$

where $a(j)$ is the unweighted acceleration response at frequency $f_j$.

3. RESULTS

Seat control is only necessary on poor surfaces. Hence poor to severe tracks were used as inputs. A road scale factor was used for the spectral density of the ground profile.

This was set at 1 for a gravel road, with a good road corresponding to a scale factor of 0.11 and a severe track to a factor of 5 (amplitudes 5 times greater than a gravel road).

A semi-active suspension was employed to reduce the bounce resonance (Figure 2) which occurs at approximately 1.4 Hz. The objective of the seat control system is to reduce the internal human body response at around 4.5 Hz. This shows up particularly in the visceral acceleration (Figure 3).

In Figure 3 the visceral displacement $x_v$ predicted by the observer is compared with the value found from assuming the system to be fully observable. For completeness the case of no seat control was included. The agreement between predicted and ‘exact’ visceral displacement is good, particularly in the case of seat control where the observer is actually needed.

It is assumed that comfort is set by visceral acceleration. This is shown in Figure 4, where the visceral acceleration in the seat control case is compared with that in the case where there is no seat control.

The passive response exhibits two peaks. The first is due to the sprung mass bounce input around 1.4 Hz (reduced but not eliminated by the suspension control system) and
Figure 3: Performance of Observer

Figure 4: Visceral Acceleration at 6 m/s; Surface Factor 5
the visceral resonance itself around 4.5 Hz. Ideal (fully observable) control eliminates the visceral acceleration peak but is not so much better than the observer in controlling the resonance around 1.4 Hz (although this has been virtually halved by suspension control (see Figure 2).

The total discomfort is related to $A_{TOT}$, (section 2.5), obtained summing the mean square acceleration at each frequency, and multiplying by the weighting at each frequency. This is shown in Figure 5 for vehicle speeds of 4 m/s to 8 m/s over rough terrain (surface factor 5). Ideal control would achieve a reduction of around 65% relative to the case of no seat control. With the observer the reduction is 39%.

4. CONCLUSIONS

The observer proposed - a simple and practical one – achieves about 60% of what could be achieved in a fully observable system. This appears acceptable at the present stage of development.

The performance of the observer needs to be tested in an experimental system, and this is one of the next objectives. A second objective is to pursue an improved observer. In an experimental set up seat and sprung mass accelerations could be measured using accelerometers, which would allow relative displacements to be obtained without the use of a pull-wire transducer, which might not be robust enough for in car, off road, application. Test work to date in the laboratory resulted in a fatigue failure of the transducer.

5. ACKNOWLEDGEMENTS

Thanks are due to the Royal Society of London for their support of the collaborative programme between the Department of Mechanical Engineering at the University of Bath and the Institute of Solid Mechanics in Bucharest, which made this research possible, and also to Sarah Fuge for her assistance with the preparation of the paper.

6. REFERENCES

IMPROVING THE MANUFACTURE OF EXTRUDED WINDOW SEALS VIA SIMULATION

Marcelo Zottolo
Edward Williams
Production Modeling Corporation
Suite 1006 Parklane Towers West
Three Parklane Boulevard
Dearborn, Michigan 48126 U.S.A.

ABSTRACT

Discrete-event process simulation, which boasts a long and enviable history of guiding improvements to manufacturing operations, extended its successes in the study reported here. A small manufacturer of extruded window seals, as a member of the automotive industry supply chain, used discrete-event simulation and allied statistical input and output analyses to comparatively evaluate four alternatives and select the best one based on various performance metrics and business scenarios.

INTRODUCTION

Discrete-event process simulation has helped improve efficiency and economy of operations in many fields, such as manufacturing, health-care, service, transportation, and logistics; among these, manufacturing applications are the most venerable and perhaps still the most numerous (Miller and Pegden 2000). Examples of such studies in the literature are numerous. For example, (Giribone, Mosca, and Queirolo 2003) applies simulation to validation of weekly production plans. (Palacios 2003) describes the application of simulation to integrate systems, both vertically and horizontally, to operations such as planning, soaking, sorting, and kiln-drying in the lumber industry. (Carpanzano and Cataldo) 2003 applies simulation to verify logic controls for agile manufacture of shoes. Again within the automotive industry’s far-reaching supply chains, (Türkseven and Ertek 2003) uses simulation to improve both the quality and the efficiency of manufacturing steel cord used to fabricate radial tires. In the current study, a supplier of automotive parts teamed with an industrial-engineering consulting company to conduct a study which provided the supplier with valuable information concerning performance metrics under each of four scenarios.

To describe this study, the authors will first provide a descriptive overview of the manufacturing process, emphasizing the questions and concerns which collectively urged the client to seek both strategic and tactical guidance from a process-simulation analysis. After a description of the construction, verification, and validation of the model, this paper will then present the results of the study for the various scenarios considered, and summarize the results and their implications for future analogous studies.

OVERVIEW OF MANUFACTURING PROCESS AND ITS CONCERNS

The client in question manufactures rubber window seals for trucks ultimately sold to private consumers (not overland trucking companies) by a large automotive manufacturer. These seals are made in six varieties: left-hand versus right-hand, and front-window (“quad” or standard) versus rear-window. Hence, in terminology typical to the economic region surrounding Detroit, Michigan, United States, the client is a “first tier automotive supplier.” Specifically, the production line wherein this client sought guidance and improvement via simulation analyses was an extrusion line; extrusion being a generic term for a manufacturing process in which a billet of raw material (rubber in this case), enclosed in a container, is forced through an aperture whose cross-sectional area and dimensions are smaller than those of the original billet (El Wakil 1998). After extruding raw generic lengths of outer belt seals and cutting them to specified lengths, the manufacturing process must first send them to a work-in-process [WIP] cooling area, where they cool for at least one hour in a water quench bath before application of glue, attachment of flock (soft fuzzy material) to the glue, and cutting to length pertinent to the variety currently being produced. Production is in batch mode – i.e., only one product variety (length) is in production in this line at any particular time. The parts are then transported to assembly machines which punch them and assemble end caps automatically. These assembly machines then disgorge the assembled parts to an inspection area. There, an operator inspects them and packs satisfactory ones in containers for shipment.

Specific performance metrics of high concern were WIP level, safety-stock levels, throughput, and order-to-delivery elapsed time under a variety of demand scenarios. Client management knew from experience with the highly volatile consumer truck market that some safety stock needed to be kept in inventory, despite the associated carrying costs, as a hedge against sudden high-urgency demands from their key customer, the automotive manufacturer. Since these managers held discretionary decision-making
authority over numerous parameters: line speeds, cycle times, number of operators, number and length of work shifts, etc., they specified that the simulation model be able to explore alternatives pertinent to all of these. In particular, they faced impending highly committal decisions involving both facility reconfiguration to accommodate huge machines and WIP storage areas, and the potential hiring of additional production employees.

Additionally, the client engineers had advocated use of simulation at their company previously; this study represented the first budgetary approval of its use. Therefore, it was important to them to achieve a conspicuous success with this project to establish a beachhead for ongoing beneficial use of simulation within the company (Williams 1997). In this context, “conspicuous success” explicitly included strict adherence to stern budgetary constraints on both financial and time expenditures.

MODEL CONSTRUCTION, VERIFICATION, AND VALIDATION

Having decided on the scope of the model and analysis (the batch-mode extrusion line), client engineers and the simulation analysts next collaborated to specify fundamentals of model construction. For example, the production schedule used for the base model comprised four 24-hour days per week, in three shifts (7am to 3pm, 3pm to 11pm, and 11pm to 7am). Each shift included two breaks of fifteen minutes each, and operators were assumed always available except during those breaks. Likewise, raw material was assumed always available. Shipment of completed work (hence vacating output WIP areas) occurred daily at noon. Animation would be two-dimensional only. As simulation experts have repeatedly remarked (Sadowski 1991), documentation and mutual (client-analyst) understanding and acceptance of assumptions such as these is essential, lest the client unwittingly misuse, misinterpret, or unjustifiably extrapolate results later obtained from the model.

Next, the client engineers and consulting analysts discussed the simulation software of choice; the discussions incorporated many of the ideas and methods recommended in (Tewoldeberhan et al. 2002). In view of the fiscal constraints pertaining to the study, the absence of need to model detailed material-handling systems (such as, for example, fleets of automatic guided vehicles), and the acceptability of two-dimensional animation, the project team agreed upon the use of SIMUL8® software. This software package and simulation tool permits simultaneous construction of a model and its two-dimensional animation, and provides basic constructs such as Work Centers, Storages (buffers and/or queues), Resources (e.g., machine operators or special pieces of equipment), Conveyors, Work Entry Points, and Work Exit Points (Hauge and Paige 2001). One unexpected difficulty in building the model was the difficulty of representing two distinct shift patterns for the facility – one for weekdays and another for weekends. It is straightforward in SIMUL8® to specify a shift pattern to control availability of resources, and also to specify that a shift pattern controls only a specified subset of days of the week. However, there is no canonical way of defining two contrasting shift patterns and transferring “control” from either to the other as the simulation clock advances among the days of the week. The modeling team overcame this difficulty by defining a shift pattern for the weekdays and then writing code in Visual Logic (SIMUL8®’s internal programming language) to control resource availability on weekends. An additional limitation of SIMUL8® is its inability to distinguish, in its reports of performance statistics, whether a work center is on- or off-shift. For example, if a work center is idle when it goes off-shift, the off-shift time is reported as idle time; if a work center is busy when it goes off-shift, the off-shift time is reported as busy time. This hindrance to precise interpretation of output statistics relative to machine utilization was overcome by a small amount of Visual Logic coding within the model plus routine computations “downstream” within the Excel® worksheets receiving SIMUL8® output.

Construction of the model incorporated, in addition to the basic process flow through a variety of operations, downtimes, setup times, and usage of labor, including shift patterns. The model construction proceeded concurrently with gathering of additional data, such as downtime data, by the client engineers. In view of uncertainties acknowledged relative to many of these data, the engineers and analysts agreed that (a) extensive sensitivity analyses would be undertaken, and (b) input data parameters would be extensively incorporated into a spreadsheet which client engineers could conveniently revise during experimentation. In particular, the availability of the inventory level, updated in “real simulation time” within an Excel® worksheet as the model ran, provided an unexpected nicety of high value to the client engineers and their managers. Model verification and validation techniques used included exhaustive examination of the animation, structured walkthroughs, extreme condition tests (for example, an experimental run in which no or all parts are rejected at an inspection point), examination of stepwise model traces, and Turing tests (Sargent 2003). Significantly, team meetings involved not only client engineers, but also their managers – a team effort vital to the credibility of the simulation and the eventual incorporation of its results into client operational practice (Scheeres 2003).
# RESULTS OF THE STUDY

Four different scenarios, summarized in the tables below, were run for comparative analysis.

<table>
<thead>
<tr>
<th>Scenario #1.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Monday thru Thursday</strong></td>
</tr>
<tr>
<td>SHIFT 1 7:00 - 3:00</td>
</tr>
<tr>
<td>Extrusion</td>
</tr>
<tr>
<td>SHIFT 2 3:00 - 11:00</td>
</tr>
<tr>
<td>Extrusion</td>
</tr>
<tr>
<td>SHIFT 3 11:00 - 7:00</td>
</tr>
<tr>
<td>Extrusion</td>
</tr>
</tbody>
</table>

**Summary**

<table>
<thead>
<tr>
<th>Line</th>
<th>Hours/Week</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extrusion</td>
<td>96</td>
</tr>
<tr>
<td>Assy MC 1</td>
<td>60</td>
</tr>
<tr>
<td>Assy MC 2</td>
<td>80</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario #2.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Monday thru Thursday</strong></td>
</tr>
<tr>
<td>SHIFT 1 7:00 - 3:00</td>
</tr>
<tr>
<td>Extrusion</td>
</tr>
<tr>
<td>SHIFT 2 3:00 - 11:00</td>
</tr>
<tr>
<td>Assy MC 2 (2nd half only), Assy MC 2 (2nd half only)</td>
</tr>
<tr>
<td>Extrusion</td>
</tr>
<tr>
<td>SHIFT 3 11:00 - 7:00</td>
</tr>
<tr>
<td>Extrusion</td>
</tr>
</tbody>
</table>

**Summary**

<table>
<thead>
<tr>
<th>Line</th>
<th>Hours/Week</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extrusion</td>
<td>96</td>
</tr>
<tr>
<td>Assy MC 1</td>
<td>60</td>
</tr>
<tr>
<td>Assy MC 2</td>
<td>60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario #4.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Monday thru Thursday</strong></td>
</tr>
<tr>
<td>SHIFT 1 7:00 - 3:00</td>
</tr>
<tr>
<td>Assy MC 2 (2nd half only)</td>
</tr>
<tr>
<td>SHIFT 2 3:00 - 11:00</td>
</tr>
<tr>
<td>Assy MC 2 (2nd half only)</td>
</tr>
<tr>
<td>SHIFT 3 11:00 - 7:00</td>
</tr>
<tr>
<td>Assy MC 2 (2nd half only)</td>
</tr>
</tbody>
</table>

**Summary**

<table>
<thead>
<tr>
<th>Line</th>
<th>Hours/Week</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extrusion</td>
<td>96</td>
</tr>
<tr>
<td>Assy MC 1</td>
<td>48</td>
</tr>
<tr>
<td>Assy MC 2</td>
<td>48</td>
</tr>
</tbody>
</table>
Each scenario was run with a warm-up of 220 hours, a run length of 960 hours, and three replications. The long warm-up and run times were attributable to the rare but significantly long equipment downtimes in the model; achieving steady state and obtaining representative results within each replication depended on allowing each downtime to occur at least five or six times (Harrell and Tumay 1995). In contrast, the low number of replications was attributable to the relatively small amounts of stochastic variation within the model. Cycle times were fixed; the only variability in the model appeared in the rarely used breakdown parameters (time to failure and time to repair). For Scenario #1, key results were:

a) Based on stated assumptions, demand can be fully satisfied.

b) Overall, assembly crews (two for each of shifts one and two) are 23% idle due to lack of work-in-process (WIP) seals to process.

c) Daily, the second shift starts with the work-in-process (WIP) area nearly empty.

d) The extrusion line paces the system and assembly crews are busy half of the shift.

For Scenarios #2 and #3, key results were:

a) Based on stated assumptions, demand can be fully satisfied

b) WIP area runs out of seals by Friday at the beginning of the second shift, implying no assembly production during the second shift (Scenario #2) or during both the second and third shifts (Scenario #3).

c) Assembly lines are idle approximately 8% of the time.

d) Scenario #3 has less average WIP inventory than does Scenario #2.

e) Since production exceeds demand, the finished seals inventory grows without an endogenous bound.

For Scenario #4, key results were:

a) Based on stated assumptions, demand can be fully satisfied.

b) The WIP area never runs out of parts; therefore, both assembly lines always have WIP seals to process and their idle time is zero.

c) Since production exceeds demand, the finished seals inventory grows without an endogenous bound. However, maximum levels of WIP reached are lower than those of either Scenario #2 or Scenario #3.

**SUMMARY AND CONCLUSIONS**

At project inception, the optimum choice of analytical tool(s) to solve the client’s problem was quite unclear; the natural candidates were basic spreadsheet (Excel®) analysis, use of a discrete-event process simulation tool, and/or use of scheduling analysis. Initial discussions with the client exposed the tacit assumption that schedules would change infrequently – at least within the near-term time frame (approximately one calendar quarter) within which the clients and their managers were eager to obtain efficiency improvements. Therefore, the possibility of using scheduling analysis and its software tools were set aside for potential later use; indeed, such analysis is now a likely aftermath of the work described in this paper. The sheer problem complexity defied the possibility of using spreadsheet analyses alone; rather, spreadsheets were used, as noted above, as a convenient milieu for data input and results output from discrete-event simulation.

The clients and their managers, well satisfied with the timeliness and unequivocal results of this simulation study, used it as a basis for deciding: among the four scenarios presented above. Specifically, they fixed upon Scenario #4, which, like the first three scenarios, satisfied market demand and additionally offered the following advantages:

a) The assembly lines were never idle.

b) The work-in-process area never suffered stockouts.

c) Due to better balancing of labor resources between the extrusion and assembly departments, no more crews were necessary—a significant economic advantage.

Furthermore, this successful application of simulation in this business environment originally having little familiarity with it has increased the likelihood of its future use as an analytical tool within the client company.

**ACKNOWLEDGMENTS**

The authors gratefully acknowledge the assistance of Rohit Trivedi, a professional colleague and educator in simulation (the latter at Wayne State University, Detroit, Michigan, United States) in suggesting improvements to this paper.

**REFERENCES**


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 December 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 classes at the University of Michigan, including both undergraduate and graduate simulation classes using GPSS/H®, SLAM II®, SIMAN®, ProModel®, SIMUL8®, or Arena®. 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 Monterey, México; Istanbul, Turkey; Genova, Italy; and Riga, Latvia. He served as a co-editor of *Proceedings of the International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation 2003*, a conference held in Riga, Latvia.

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>
SPATIAL MULTIBODY MODELING AND VEHICLE DYNAMICS
ANALYSIS OF A FUTURE MILITARY TRUCK SYSTEM CONCEPT

Michael D. Letherwood, David D. Gunter, David J. Gorsich, Jose Mabesa, Jr.
National Automotive Center (NAC), US Army Research, Development & Engineering Command (RDECOM)
Warren, Michigan 48397-5000

Keywords: Future Tactical Truck Systems, Future Combat Systems, Vehicle Dynamics, Stability, Ride Quality

Abstract
The US Army vision, announced in October of 1999, encompasses people, readiness, and transformation. The goal of the Army vision is to transition the entire Army into a force that is strategically responsive and dominant at every point of the spectrum of operations. The transformation component will be accomplished in three ways: the Objective Force, the Legacy (current) Force, and the Interim Force. The objective force is not platform driven, but rather the focus is on achieving capabilities that will operate as a “system of systems.” As part of the Objective Force, the US Army plans to begin production of the Future Combat System (FCS) in FY08 and field the first unit by FY10 as currently defined in the FCS solicitation[1]. As part of the FCS program, the Future Tactical Truck System (FTTS) encompasses all US Army tactical wheeled vehicles and its initial efforts will focus only on the heavy class. The National Automotive Center (NAC) is using modeling and simulation to demonstrate the feasibility and operational potential of advanced commercial and military technologies with application to new and existing tactical vehicles and to describe potential future vehicle capabilities.

This document presents the results of a computer-based, vehicle dynamics performance assessment of an FTTS concept with such features as a variable height, hydraulic, trailing arm suspension, skid steering, and in-hub electric drive motors. A fully three-dimensional FTTS model was created using a commercially available modeling and simulation methodology and validated through studies were performed by comparing model predictions with baseline, validated model predictions from another vehicle in the same size and class as the FTTS concept vehicle. The model was considered accurate enough to predict various aspects of ride quality and stability performance, critical to US Army Objective Force mission needs. One-to-one comparisons of the FTTS and a standard, solid-axle, Heavy Tactical Vehicle (HTV) operating in various terrain and obstacle negotiation conditions were performed.

The objective of this paper and presentation will be to describe how M&S is being applied to answer a wide variety of design and performance evaluation questions and will depict a series of simulation-based engineering projects that build on the Army's simulation investments as a tool to investigate and answer real-world vehicle design, acquisition, and engineering support questions. Due to much increased HPC computational speeds, memory, and asset availability, entire spectrums of operational mission scenarios are investigated and simulations conducted over a wide range of vehicle speeds and operating conditions. Recent major upgrades in HPC facilities now allow the highly detailed, computationally intensive models to be run in a fraction of the time, and, more importantly, many more 'what if' studies are being performed. Using HPC-based vehicle performance modeling & simulation in support of acquisition allows the Army to evaluate the performance of numerous proposed vehicle system configurations analytically, thereby saving time and costs associated with building and testing actual prototypes. The NAC’s M&S efforts using HPC is constantly striving to make the Army a smarter and more cost-effective buyer of equipment, and more importantly, significantly reducing the associated risks that are inherent in procuring newly designed, untested equipment.

INTRODUCTION
The NAC serves as the Army's agent for advancing the development of dual-use automotive technologies by industry, academia and the military services. By cultivating relationships and forming cost-shared partnerships, the NAC accelerates the exchange and implementation of automotive technologies creating developmental savings that are shared by all participants. The U.S. Military requires flexible, effective and efficient multi-mission forces capable of projecting overwhelming military power worldwide. To satisfy this requirement, the joint Army/DARPA FCS program was developed to provide enhancements in land force lethality, protection, mobility, deployability, sustainability, and command and control capabilities. The goal of the FCS program is to design an ensemble that strikes an optimum balance between critical performance factors, including ground platform strategic, operational and tactical mobility; lethality; survivability; and sustainability. This “system of systems” design will be accomplished by using modeling and simulation and experimentation to evaluate competitive concepts as defined in the FCS simulation support plan[2]. The FCS will be capable of adjusting to a changing set of missions, ranging from warfighting to peacekeeping, as the deployment unfolds.
What sort of tactical wheeled vehicles will be needed to support the Army in the next decade? To provide accurate answers to that question, the NAC created the FTTS concept (see figure 1) to investigate critical technologies that will be required to achieve stated FCS goals and objectives. Some of the technologies to be explored in this analysis to enhance stability, handling and mobility include; active-variable height hydropneumatic suspensions, advanced hybrid electric propulsion systems, electronic steering, and central tire inflation.

**Figure 1.** FTTS Concept Vehicle Representation^{3}

**FTTS CONCEPT REQUIREMENTS**

NAC engineers are using three-dimensional illustrations, models, simulations and various other analyses to help optimize the survivability, mobility and supportability of the Army’s future tactical trucks. A combination of virtual prototypes and critical hardware demonstrations will leverage both the Army’s technology programs and those from the commercial sector to revolutionize future logistics support. The US Army Objective Force mission requirements will include a heavy tactical vehicle capable of transporting 11 ton payloads on and off road. With armor modules and payload included, this FTTS concept vehicle will have a gross vehicle weight (gvw) which exceeds 25 tons (see figure 2).

**Figure 2.** FTTS Concept Vehicle Representation^{3}

It will also feature a palletized load handling system and be capable of transporting 4 and 8 foot ISO containers. The operating requirements for this vehicle are expected to be much more severe than equivalent vehicles in today’s fleet. Due to the lighter overall system weights and increased performance, the FTTS will be required to travel 65 miles per hour (mph) on road and 30 mph off road. The 8x8 FTTS concept has 8 independent, active, in-arm variable height, hydropneumatic trailing arm suspensions which provide approximately 18 inches of wheel travel. It also has 8 in-hub permanent magnet, electric drive motors for vehicle system propulsion that runs silent and can provide braking. The FTTS vehicle maintains an advanced, lightweight, 450 horsepower diesel engine to keep the batteries charged and provide power to the system as required during operation. Lastly, the system has large 16xR20 radial tires with run flat inserts and central tire inflation (CTIS) to maximize off road mobility and payload carrying capacity. The main factors that were analyzed during this analysis to evaluate the stability, handling, ride-quality, and mobility characteristics of the FTTS concept.

**APPROACH**

NAC engineers used a commercially available vehicle modeling and simulation methodology called DADS^{6}, or Dynamic Analysis and Design Systems, to generate and simulate models of the FTTS concept. This model would be suitable for obtaining a better understanding of the vehicle’s performance characteristics and for assessing future technology upgrades that could lead to better performance. The FTTS model includes accurate representation of all suspension components to provide adequate predictions of relative displacement between subsystem components. Nonlinear models of suspension stiffness and damping, jounce and rebound stops and steering stops were incorporated to provide accurate interaction force predictions. Individual rotating wheels with nonlinear tire/terrain interaction models that allow the wheels to leave the surface were included in the model to allow large vehicle displacements, including rollover. Rolling tire models that generate fore-aft and side-to-side tractive forces between tire and terrain were included to insure representative mobility predictions. Second order steering and speed control algorithms were used to keep the vehicle model on course and to maintain desired speed based on electronic steering curves and electric motor output curves, respectively.

The FTTS model, loaded to 11 tons, was executed over a number of artificial pothole and bump obstacles defined by NAC engineers. The purpose of these short duration, transient maneuvers was to provide repeatable disturbance inputs to the model with well controlled initial conditions. The model was also executed over a number of straight-line symmetric and non-symmetric variants of the Perryman 2 and 3, and Churchville B courses located at US Army test sites. The simulations were conducted at various speeds to induce different levels of response, and to investigate the upper limits of safe operational performance. A similar model of the 10-ton heavy tactical vehicle (see figure 3) was developed under a different project.
This model, loaded to 10 tons, was executed over the same obstacles, maneuvers and courses as the FTTS, and at the same speeds so one-to-one comparisons could be made between the two vehicle systems. Side-by-side computer-generated animations of each simulation were made and recorded on video tape for review in real time. Comparison of the results indicated superior FTTS performance over the HTV in all simulations performed. In a number of cases, the HTV showed violent motion or rolled over, while the FTTS showed much less severe motion and remained upright. Based on the above comparisons described in this report, the FTTS appears to be substantially more stable than the HTV, and it should have a higher probability of meeting US Army Objective Force mission requirements.

This paper first describes how a representative FTTS model was defined within the limits of the time and cost budget, and how a representative model was developed. A general overview is first provided of the topology, parameters and performance characteristics of each major subsystem. Then we provide an assessment of how each subsystem might influence the vehicle’s operating performance envelope. Details of the 10 ton HTV model are given in (5) and are not presented here. A general overview of the modeling strategy and a description of each major vehicle subsystem model are given to provide a better understanding of the composite vehicle model operation, its interaction with the obstacle and terrain models, and the simulation results. Data and descriptions of each obstacle, maneuver and course profile are presented. The matrix of obstacle negotiations, maneuvers and course negotiations is presented, and simulation results are summarized.

**Acquisition of Data for The FTTS Model**

Figure 4 shows a computer generated graphical representation of the FTTS model which contains the eight by eight FTTS suspension and 11 ton chassis.

**FTTS Body Structure and Model**

As shown in figure 5, the chassis contains a rigid ladder frame along the full length of the vehicle that each of the trailing arm units attach to along with the corresponding wheel hub and electric motor assemblies. Each trailing axle pivots up and down to allow the necessary suspension displacement with a jounce and rebound stop installed on each axle to prevent excessive suspension travel. All axles have hydropneumatic arrangements to provide the necessary suspension support and damping. Skid steering is achieved by electronically controlling the electric motors in each wheel to control the wheel speeds to steer the truck. The FTTS steering arrangement allows the wheels to be...
electronically controlled and is performed by a 2nd order non-linear algorithm.

![FTTS Graphical Representation](image)

**Figure 5. FTTS Graphical Representation**

A brief kinematic description of the model implementation of the FTTS is now presented. The vehicle chassis, including load is represented by a single rigid body. Eight rigid bodies represent the eight trailing axles, and are connected to the chassis body by revolute joints which are aligned with them. Eight additional rigid bodies were used to represent the hydropneumatic springs which are connected to the chassis by transverse revolute joints at the physical pivot point locations in the vehicle. The remaining bodies are eight wheels connected to the trailing axles and eight wheel hubs connected to the axles. The wheels are connected by transverse revolute joints and the wheel hubs are connected by revolute joints aligned along the wheel axes.

**FTTS Suspension, Tire, Powertrain and Steering Models**

The vehicle contains a number of hard and soft mechanical stops that must be adequately modeled to insure proper inter-component displacements. A nonlinear translational spring between the chassis and each trailing axle was used to represent the corresponding jounce stops. These springs were placed in line with the physical stops mounted on the chassis, and the hardening rates of the corresponding hydropneumatic springing curves were modeled with nonlinear functions. An additional pair of similar nonlinear springs were placed in line with the rebound stops on the axles. Appropriate metal-to-metal rotational stops to prevent oversteering the wheels was also included. These stops were represented by very stiff nonlinear rotational springs around the attachment revolute joint. In addition, each wheel hub has a rotational stop to limit maximum rotation relative to the axle. These hard stops are represented by stiff nonlinear rotational springs around the corresponding wheel axes. Compliance in the hydropneumatic suspension was represented by connecting vertical translational springs between the chassis and the center of the corresponding trailing axle bodies. The translational spring rates were set equal to the effective hydropneumatic stiffnesses in loading and unloading. Damping was also included in each suspension unit.

The tires were modeled by nonlinear springs that allow the interaction forces to go to zero when they leave the ground. The vertical force developed between the FTTS tires and a non-deforming ground surface as a function of vertical tire deflection was taken from tests performed on this tire. Additional data causing the curve to increase nonlinearly to emulate hardening effects due to bottoming out on the rims was added. Also, the output force is zero for all negative displacements. The vertical stiffness rate corresponds to a cross-country tire inflation rate. The tire model also computes the relative slip velocity between the bottom of the tire and ground. This slip velocity is used to compute a fore-aft slip between zero and 100 percent which is inserted into the longitudinal friction curve. The coefficient of friction from this curve is then multiplied by the normal force to obtain the longitudinal frictional force. This force always acts on the tire in the opposite direction of the relative slip. A lateral slip angle in radians is also computed from the two components of forward and lateral slip velocity and inserted into the curve. To find lateral tire forces for vertical forces, the curves are interpolated with vertical tire force and slip angle values to obtain lateral force magnitudes and directions. These forces are applied to the wheel bodies to support the vehicle and control it. All tire data used here is given in. The wheels are driven by a speed control algorithm. A desired constant or variable speed control signal based on the in-hub electric motors as input to the model is used as a reference. The speed of the vehicle is determined by projecting its velocity vector along the chassis fore-aft centerline. This result is compared to the desired speed and a corrective torque is generated. This torque is applied to each wheel to propel or brake the corresponding wheel, which effectively controls the vehicle motion. A simple steering algorithm monitors the vehicle’s centerline alignment with, and its deviation away from a designated trajectory. These two error signals are converted into a steering torque that is applied at each wheel to steer the vehicle. The gain in this controller model was made inversely proportional to vehicle speed to reduce steering sensitivity at higher speeds for better steering stability.

**TEST SCENARIOS**

The FTTS model, loaded to 11 tons, and the HTV model loaded to 10 tons were executed over a number of defined artificial pothole and bump obstacles. Each obstacle was set up so only the left side tires encountered it in order to induce significant nonsymmetrical transient responses. A
number of lane change and slalom maneuver simulations were also conducted. The purpose of these short duration, transient maneuvers was to provide repeatable disturbance inputs to the model with well controlled initial conditions. The model was also executed over a number of straight-line symmetric and nonsymmetric variants of the Perryman 2 and 3, and Churchville B courses. The simulations were conducted at various speeds to induce different levels of response, and to investigate the upper limits of safe operational performance. The following computer-based simulations were set up primarily to determine trends and investigate upper stability limits of the vehicle systems:

**Cross Country Courses**

Vehicle models were set up to negotiate 180 foot sections of straight-line representations of portions of measured left and right track elevation profiles of Churchville B, Perryman 2, and Perryman 3 course as functions of distance traveled along the course were used. The two tracks are assumed to be 6 feet apart corresponding to the approximate track width of the vehicles. Each of these courses was also modified by shifting the left track 9 feet ahead of the right track to simulate non-symmetrical terrain inputs to the vehicles. Results obtained from the vehicle simulations on these course segments should not be taken as indications of how the corresponding vehicles would perform on the actual courses because these courses also contain hills, curves and soil properties not included in the models. The results should be used primarily for making comparisons between the two vehicles as they would perform on these artificial course segments.

**Lane Change Maneuvers**

This set of simulations emulates a double lane change/obstacle avoidance maneuver. This set of simulations emulates an obstacle avoidance or slalom maneuver. Each maneuver executes a lateral transition to the left and a reverse transition back to the original lane.

**Single Bump Negotiation**

This set of simulations emulates the negotiation of artificially constructed ramps and potholes. The simulations were performed on various ramp heights and pothole depths and only the left side wheels of the truck encounter the disturbances.

**VARIABLES RECORDED AND SIMULATIONS CONDUCTED**

In order to determine the FTTS’s stability characteristics, we included several cross country courses in the test matrix. These cross country courses vary in roughness and also in the amount of roll they induce. In addition to these cross country courses, the test matrix included lane change and slalom maneuvers. These maneuvers are instrumental in determining a vehicle’s Lateral Acceleration Threshold (LAT). The LAT is the highest lateral acceleration a vehicle can withstand without rolling over. Because we were performing comparisons between vehicles, rather than tabulating a list of lateral accelerations, we ran the lane change simulations at speeds high enough to cause one vehicle to roll over. In both cases, the HTV rolled over at 64 km/h and the FTTS remained upright. The test matrix also included several pothole and bump obstacle courses. These courses excite the vehicle system with a single discrete event which enables us to easily compare the effects of suspension jounce clearance by measuring the percent of time on jounce stops. Though not as pronounced as the cross country courses, the courses do impart significant roll and pitch motion to the vehicle system, thereby allowing us to compare vehicle roll and pitch compliance. Table 1 gives a summary of the different simulations performed.

<table>
<thead>
<tr>
<th>File names</th>
<th>Course Description (m-meter)</th>
<th>Speeds (km/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>chvb16,32,48</td>
<td>Churchville B - 152 m</td>
<td>16, 32, 48</td>
</tr>
<tr>
<td>pm216,32,48</td>
<td>Perryman 2 - 152 m</td>
<td>16, 32, 48</td>
</tr>
<tr>
<td>pm2os16,32,48</td>
<td>Perryman 2-152 m</td>
<td>16, 32, 48</td>
</tr>
<tr>
<td>pm316,32</td>
<td>Perryman 3-152 m</td>
<td>16, 32</td>
</tr>
<tr>
<td>pm3so16,32,48</td>
<td>Perryman 3-152 m</td>
<td>16, 32, 48</td>
</tr>
<tr>
<td>lane32,48,64</td>
<td>lane change-3.3x27.4 m</td>
<td>32,48,64</td>
</tr>
<tr>
<td>.2286ph32,25</td>
<td>pothole-.2286 m deep</td>
<td>32,25</td>
</tr>
<tr>
<td>.3048ph16,32,40,48</td>
<td>pothole-.3048 m deep</td>
<td>16,32,40,48</td>
</tr>
<tr>
<td>.4572bp24,32</td>
<td>bump-.4572 m high</td>
<td>24,32</td>
</tr>
<tr>
<td>.6096bp16</td>
<td>bump-.6096 m high</td>
<td>16</td>
</tr>
</tbody>
</table>

**Table 1. Simulations Performed on Vehicles**

The FTTS and HTV simulations are most suitable for analyzing ride quality, and to a lesser extent for quantifying vehicle stability. To help obtain qualitative assessments of ride quality in the two vehicles, a number of accelerations, and force and torque time histories of both vehicle systems were plotted. Because the two vehicles have different suspension designs, one cannot always make one-to-one comparisons between amplitudes of corresponding variables. In some cases, it may be necessary to compare relative magnitudes within the particular plots. For example, in the FTTS an axle’s angle is a direct indication of the corresponding suspension displacement, whereas, in the HTV an axle’s spring displacement indicates the corresponding suspension displacement.

**RESULTS AND DISCUSSION**

Due to limited space requirements, results will be summarized in the following sections and in the conclusions. Plots of selected FTTS and HTV state variables for the simulations performed on the 152 meter cross country course segments, lane changes, pothole and bump negotiations, and power spectral density computations on
the vertical component of chassis translational acceleration, and the pitch and roll components of chassis rotational acceleration at the vehicle speeds indicated in Table 1 will be discussed. When performing a computer-based vehicle analysis, we generally compare the vehicle’s response with a ‘similar’ vehicle. We simulated the HTV executing the same test matrix as used for the FTTS. Evaluating the FTTS becomes a matter of comparing state variables that lend insight to vehicle stability and ride quality characteristics.

Stability Analysis - Roll Stability Comparisons

A major stability concern is vehicle roll compliance. Roll compliance is the ability of a vehicle suspension to absorb terrain roll undulations without imparting them to the sprung mass. To quantify the roll compliance, we recorded the chassis roll angle and the roll acceleration for each vehicle and processed these data to determine the maxima, minima and root mean square (RMS) values. When a vehicle rolls over or the motion becomes extremely violent, the accelerations and angles can become so large that they dwarf the other data samples. This makes it difficult to compare the other good data. To avoid this problem, we replaced the exceptionally large positive values by small ‘negative’ values in the bar charts so they could be readily identified as representing ignored data points. A bar chart comparing both vehicles while traversing the cross country courses is shown in Figure 6. With the exception of the HTV rollovers and the near-rollover on Churchville B at 48 km/h, the RMS roll angles of both vehicles are comparable with the FTTS values averaging slightly less. In a similar manner, the vehicle roll accelerations are comparable with the FTTS roll accelerations being significantly less.

Stability Analysis - Pitch Stability Comparisons

Pitch compliance is an indicator of a suspension’s ability to absorb terrain pitch undulations without imparting them to the sprung mass. To quantify pitch compliance we recorded the chassis pitch angle and pitch acceleration for each vehicle and processed these data to determine the maxima, minima and RMS values for both vehicles on the cross country courses. Again, small negative entries identify HTV rollover or near-rollover events. A bar chart comparing both vehicles while traversing the cross country courses is shown in Figure 7. With the exception of the HTV rollover and near-rollover cases, the RMS pitch angles are comparable with the HTV angles being generally higher. Except for HTV rollover and near-rollover cases, the RMS pitch accelerations are comparable. However, when the pitch accelerations are small, the HTV does better, but the FTTS does better when the pitch accelerations are large.

Stability Analysis - Percent Time Tire Airborne

For the driver to maximize vehicle stability, the foremost rule is to keep the tires on the ground as much as possible. To determine the ability of each vehicle to do this, we computed the percent of time each tire was off the ground. To facilitate comparison between vehicles, we averaged these percentages for each simulation to produce a percentage airborne-time. As above, small negative bar chart entries denote HTV rollover or near-rollover events. In general, the FTTS performed much better than the HTV in this situation. A bar chart comparing both vehicles while traversing the cross country courses is shown in Figure 8.

Ride Quality

Vertical Acceleration at Chassis Center of Mass - An important factor when considering a vehicle's performance is its ride quality. The ride quality is a measure of the severity of the ride and reflects the likelihood of passenger injury, component failure and payload damage. In order to compare ride quality characteristics, the vertical acceleration of the chassis center of mass was recorded. Small negative bar chart entries denote HTV rollover or near-rollover events. At the lower speeds, the vertical accelerations were comparable. At the higher speeds, the FTTS performed considerably better than the HTV. A bar chart comparing both vehicles while traversing the cross country courses is shown in Figure 9.
Percent Time in Jounce Contact

Because of the larger forces imparted by severe jounce impacts, the likelihood of component failure increases with the number events. Therefore we computed the percentage of time each jounce stop was in contact. To facilitate comparison between vehicles, we averaged these percentages for each simulation to produce a percentage jounce contact time. Small negative bar chart entries denote HTV rollover or near-rollover events. The FTTS performed much better than the HTV in all situations. A bar chart comparing both vehicles while traversing the cross country courses is shown in Figure 10.

Figure 10. HTV-light color bar, FTTS-dark color bar

Power Spectral Density (PSD)

Although the systematic counting and timing techniques outlined above are useful for determining vehicle stability and ride quality characteristics, the possibility exists that a given terrain profile at a given vehicle speed, may have a dominant spatial frequency that excites the vehicle and its suspension at some natural frequency. This could cause a vehicle/road interface resonance condition that may result in violent motion with subsequent loss of control. This resonance phenomena could also induce significant chassis and component flexure, and result in premature component failure. In general, a driver would instinctively avoid these resonance conditions by speeding up or slowing down, to move the profile's spatial frequency input to some other value that does not match the vehicle's natural frequencies. However, the driver model employed in these simulations was programmed to maintain a constant speed.

To investigate the possibility that either vehicle may have been operated near a resonant condition which would bias the results, we checked for possible terrain-induced resonances and the speeds that would cause them. To determine where vehicle/road resonances occur, we first computed PSD’s of the terrain elevation profiles corresponding to a vehicle speed of 1.6 km/h. We compared these frequencies with the natural frequencies of the vehicles for all cross country courses. The peaks in the PSD plots indicate the dominant frequencies for the courses at 1.6 km/h as shown in Table 2.

To determine the natural frequencies of the vehicle systems, PSD’s of chassis vertical, roll and pitch accelerations were computed and analyzed for all of the simulations. The bump and pothole obstacle course simulations are ideal for determining a vehicle’s natural frequencies because each imparts a single short duration impulse. Because small amount of energy in this discrete event, most of the spectral information is derived from the vehicle settling response. Consequently, the peaks in these PSD’s will indicate the vehicles’ dominant natural frequencies, depicted in Table 3.
### Churchville B - 1.6 kph

<table>
<thead>
<tr>
<th></th>
<th>Freq1</th>
<th>Mag1</th>
<th>Freq2</th>
<th>Mag2</th>
<th>Freq3</th>
<th>Mag3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>0.16</td>
<td>3980</td>
<td>0.28</td>
<td>794</td>
<td>0.050</td>
<td>316</td>
</tr>
<tr>
<td>Right</td>
<td>0.16</td>
<td>3980</td>
<td>0.28</td>
<td>794</td>
<td>0.050</td>
<td>316</td>
</tr>
<tr>
<td>Roll</td>
<td>0.16</td>
<td>.004</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pitch</td>
<td>0.16</td>
<td>0.32</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Perryman 2 - 1.6 kph

<table>
<thead>
<tr>
<th></th>
<th>Freq1</th>
<th>Mag1</th>
<th>Freq2</th>
<th>Mag2</th>
<th>Freq3</th>
<th>Mag3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>0.16</td>
<td>794</td>
<td>0.18</td>
<td>630</td>
<td>0.022</td>
<td>316</td>
</tr>
<tr>
<td>Right</td>
<td>0.16</td>
<td>794</td>
<td>0.18</td>
<td>630</td>
<td>0.022</td>
<td>316</td>
</tr>
<tr>
<td>Roll</td>
<td>0.16</td>
<td>.016</td>
<td>0.26</td>
<td>.006</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pitch</td>
<td>0.16</td>
<td>0.50</td>
<td>0.24</td>
<td>0.050</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Offset Perryman 2 - 1.6 kph

<table>
<thead>
<tr>
<th></th>
<th>Freq1</th>
<th>Mag1</th>
<th>Freq2</th>
<th>Mag2</th>
<th>Freq3</th>
<th>Mag3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>0.16</td>
<td>794</td>
<td>0.18</td>
<td>630</td>
<td>0.022</td>
<td>316</td>
</tr>
<tr>
<td>Right</td>
<td>0.16</td>
<td>794</td>
<td>0.18</td>
<td>630</td>
<td>0.022</td>
<td>316</td>
</tr>
<tr>
<td>Roll</td>
<td>0.16</td>
<td>.158</td>
<td>0.23</td>
<td>158</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pitch</td>
<td>0.18</td>
<td>0.40</td>
<td>0.16</td>
<td>0.035</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Perryman 3 - 1.6 kph

<table>
<thead>
<tr>
<th></th>
<th>Freq1</th>
<th>Mag1</th>
<th>Freq2</th>
<th>Mag2</th>
<th>Freq3</th>
<th>Mag3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>0.035</td>
<td>2512</td>
<td>0.18</td>
<td>2399</td>
<td>0.025</td>
<td>1514</td>
</tr>
<tr>
<td>Right</td>
<td>0.035</td>
<td>2512</td>
<td>0.18</td>
<td>2399</td>
<td>0.025</td>
<td>1514</td>
</tr>
<tr>
<td>Roll</td>
<td>0.020</td>
<td>0.006</td>
<td>0.022</td>
<td>.006</td>
<td>0.063</td>
<td>0.002</td>
</tr>
<tr>
<td>Pitch</td>
<td>0.035</td>
<td>0.251</td>
<td>0.30</td>
<td>0.200</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Offset Perryman 3 - 1.6 kph

<table>
<thead>
<tr>
<th></th>
<th>Freq1</th>
<th>Mag1</th>
<th>Freq2</th>
<th>Mag2</th>
<th>Freq3</th>
<th>Mag3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>0.035</td>
<td>2512</td>
<td>0.18</td>
<td>2399</td>
<td>0.025</td>
<td>1514</td>
</tr>
<tr>
<td>Right</td>
<td>0.035</td>
<td>2512</td>
<td>0.18</td>
<td>2399</td>
<td>0.025</td>
<td>1514</td>
</tr>
<tr>
<td>Roll</td>
<td>0.032</td>
<td>.631</td>
<td>0.18</td>
<td>.501</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pitch</td>
<td>0.035</td>
<td>0.158</td>
<td>0.032</td>
<td>.063</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 2. Dominant Freq. of Cross Country Courses

<table>
<thead>
<tr>
<th></th>
<th>Vert Acc Freq (Hz)</th>
<th>Roll Acc Freq (Hz)</th>
<th>Pitch Acc Freq (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTVR</td>
<td>1.1</td>
<td>1.0</td>
<td>1.1</td>
</tr>
<tr>
<td>FTTS</td>
<td>1.8</td>
<td>2.4</td>
<td>1.5</td>
</tr>
</tbody>
</table>

### Table 3. Vehicle Dominant Natural Frequencies

As noted earlier, the HTV’s smaller suspension clearance resulted in considerable jounce stop contacts. Nonlinearity caused by these impacts generated a broad spectrum of frequencies, and made it more difficult to pinpoint the natural frequency of the HTV. Therefore we looked for possible resonances by searching for anomalous maxima in the cross country PSD’s. After analyzing the PSD’s and the jounce stop impact data, we believe the vehicle rollovers in the simulations were caused primarily by jounce stop impacts rather than vehicle/road resonances.

### CONCLUSIONS

The most obvious result of this analysis was that the FTTS completed the entire test matrix without rolling over, whereas the HTV rolled over on several cross-country courses and lane change maneuvers. Assessments of the animations and data indicate that the FTTS has superior performance over the HTV in the US Army mission environments. One factor which may have contributed to improved FTTS performance is that the HTV payload may be a proportionally higher than the load on the FTTS. These may have combined to give the FTTS a lower center of mass, and the increased inertia give it lower roll, pitch and jounce natural frequencies than the HTV. These, plus the optimized damping ratios, may have combined to improve overall system performance. Another factor which may have contributed to improved FTTS performance is that the FTTS has a larger pitch radius of gyration, while the effective support distances between the front and rear of both vehicles is about the same (approximately 180 inches). The HTV pitch radius of gyration is close to 90 inches, while the FTTS pitch radius of gyration is somewhat less. The lower pitch radius of gyration, combined with the lower natural frequencies may result in better dynamic performance. These simulations showed that both vehicles performed well in the harsh operating environments. Any vehicle designed similar to the FTTS and operating in a similar payload range should exhibit excellent dynamic performance characteristics for the types of off-road operations expected in the field.

### REFERENCES


MODELING AND SIMULATION TECHNIQUES FOR U.S. ARMY GROUND VEHICLE SYSTEMS

Jose R. Mabesa, Jr., Aleksander Kurec, David Gorsich, Ph.D.
U.S. Army Research, Development and Engineering Command - Tank-Automotive Research, Development, and Engineering Center
(RDECOM-TARDEC)
AMSRD-TAR-N/157
Warren MI 48397-5000
Presenter: Aleksander Kurec
Tel: 586-574-7583 Fax: 586-574-8667
e-mail: kureca@tacom.army.mil

Abstract

Enhancements in laboratory simulation processes and techniques enables the extensive use of modeling and simulation to evaluate new vehicle systems, as well as proposed improvements and upgrades to existing vehicles to save both time and cost of future Army vehicle procurements. This paper discusses and presents RDECOTARDEC’s continuing efforts of applying state-of-the-art computer-based analytical modeling and physical simulation techniques for the development of U.S. Army combat and tactical vehicle systems. More importantly, the ongoing development of these laboratory tools and simulators aid in defining vehicle systems’ performance requirements, conduct evaluations and assessments of them. In particular, at the RDECOTARDEC Simulation Laboratory, engineers have modeling and simulation capabilities that are unique to U.S. Army applications, therefore offering accelerated experiment schedules, repeatable experimental conditions, and collection of data otherwise impossible to obtain. A sampling of recent modeling and simulation experiments conducted in the TARDEC Simulation Laboratory will be summarized in this paper.

Introduction

Recent advances in computational and laboratory modeling and simulation technologies have resulted in the ability to accurately determine the dynamic performance of complex vehicle systems and their subsystems. For example, high fidelity multi-body vehicle dynamics modeling methodologies, coupled with the use of characterization test fixtures and experiments, are used to accurately predict vehicle ride and shock quality, dynamic stability and control. The applications of these techniques support the research, development and evaluation of new systems and trouble-shooting existing vehicle problems.

These multi-body dynamics modeling techniques are also being applied in the TARDEC Simulation Laboratory to provide representative input motion disturbances to vehicle systems and/or subsystems. The Simulation Laboratory consists of various state-of-the-art simulators capable of replicating the shock and vibration environment experienced during harsh off-road military vehicle applications.

By using these unique tools, TARDEC engineers are able to evaluate many design alternatives and conduct performance trade-off analyses in order to quantify and/or optimize vehicle system performance to meet user specified operational requirements. These tools are also used to evaluate private-industry proposals, assess design alternatives and conduct trade-off analyses. By using these unique tools, TARDEC engineers are able to evaluate many design alternatives and conduct performance trade-off analyses in order to quantify and/or optimize vehicle system performance to meet user specified operational requirements. We also use these modeling and simulation tools to evaluate industry proposals to ensure that their proposed designs meet the performance requirements specified, thereby, making the U.S. Army a more educated buyer.

Vehicle System Characterization

Accurate quantification of three dimensional on and off road vehicle performance is highly dependent upon the vehicle’s mass and inertia properties, center of gravity, geometric characteristics, steering kinematics, suspension springs and dampers and other non-linear (friction, hysteresis etc.) parameters of the suspension system, tires, and tire/track soil interface. Thus, before any simulation modeling method can be applied, it is necessary to characterize the vehicle system to gather these important data. For conceptual vehicle systems, these data are obtained from Computer Aided Engineering (CAE) systems and engineering drawings. If the vehicle exists, these data are obtained experimentally using vehicle suspension and inertia characterization fixtures. Additionally, whenever possible limited field testing or experiments are conducted to collect data for validation of the analytical models developed. This involves instrumenting the vehicle system and using a portable data acquisition system subjecting it to a series of short duration, transient inputs under controlled initial conditions.
Table 1 summarizes many of the data items necessary to develop and validate a complex vehicle model.

<table>
<thead>
<tr>
<th>MEASURED CALCULATED SUSPENSION CHARACTERIZATION PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Suspension Rebound: Direction Spring Force Versus Deflection Measured At Wheel (Quasi-static)</td>
</tr>
<tr>
<td>Suspension Juicer: Direction Spring Force Versus Deflection Curve Measured At Wheel (Quasi-static)</td>
</tr>
<tr>
<td>Tire Force Versus Deflection Curve (Quasi-static)</td>
</tr>
<tr>
<td>Rebound Damping Characteristics (Dynamic)</td>
</tr>
<tr>
<td>Jounce Damping Characteristics (Dynamic)</td>
</tr>
<tr>
<td>Roll Center (Quasi-static)</td>
</tr>
<tr>
<td>Suspension Roll Moment Versus Roll Angle Characteristics (Quasi-static)</td>
</tr>
<tr>
<td>Suspension Roll Damping (Dynamic)</td>
</tr>
<tr>
<td>Auxiliary Roll Stiffness (Quasi-static)</td>
</tr>
<tr>
<td>Roll Steer (Quasi-static)</td>
</tr>
</tbody>
</table>

Table 1. Characterization Parameters.

At the TARDEC Simulation Laboratory, Engineers are currently in the process of procuring a Suspension Characterization Test Fixture (SCTF) to capture all of the data listed in Table 1. Once this newest addition to the Laboratory is delivered, it will truly complement the existing vehicle characterization fixture already in place, the VIPER (Vehicle Inertial Properties Evaluation Rig).

![Figure 1: 5 Ton FMTV on the VIPER](image)

The VIPER, (Figure 1.) is capable of accurately measuring system/subsystem inertial characteristics of trucks, trailers, and tank turrets. The VIPER is used to make accurate mass and inertia measurements including vehicle center of gravity location, mass moments of inertia in roll, pitch and yaw, roll/yaw mass product of inertia, axle loads (with and without payload), and vehicle weight. These properties are crucially important to the accurate analysis of vehicle rollover conditions, stability, mobility, transportability and durability.

The VIPER consists of two major components, the in-floor weigh station and the inertia platform (which acts as a stable pendulum). The lateral and longitudinal center of gravity locations are calculated using the weight station scale readings and the equations of equilibrium. The center of gravity height is computed by measuring the static roll angle (on the inertia platform) caused by known weights at known distances from the center of the inertial platform. The inertia properties are calculated by swinging the platform in all directions (roll, pitch, yaw), one direction at a time, measuring the period of oscillation and using that to compute the moment of inertia. The individual axle loads are combined to determine the overall vehicle weight. All of these properties can then be incorporated into vehicle dynamics models to create more accurate and realistic results. It is capable of handling vehicle systems up to 120 inches wide, up to 431 inches long, and up to 60,000 lbs. in weight to include the FMTV, HMMWV, PLS and any military vehicle such as the Bradley and M1A2 turrets with the use of its three different platform sizes and multiple pivot point height configurations. Mass properties, measurements using the VIPER are essential for developing realistic models used in the laboratory for physical simulation experiments.

TARDEC Simulation Laboratory Overview

TARDEC’s Simulation Laboratory utilizes a series of unique state-of-the-art motion base simulators to replicate harsh off-road vehicle motion disturbances. Physical simulation involves emulating real-time physical motions of actual vehicle systems and subsystems in a computer-controlled laboratory environment to approximate a vehicle’s field performance. At TARDEC, adaptable motion base simulators are capable of “shaking” complete combat and tactical vehicle systems weighing up to 45 tons are used to evaluate various issues associated with man and machine interaction dynamics, reliability and other component integration issues associated with the soldier, displays and controls. Physical simulation offers accelerated test schedules, repeatable test conditions, and allows collection of data otherwise difficult or impossible to obtain. Physical simulation is used to validate analytical simulation models, address soldier-in-the-loop issues, and determine failure points of a vehicle system or subsystems. There are three basic types of motion base simulators. They (1): are tire/track-coupled simulators, (2) spindle coupled simulators, and (3) platform simulators. TARDEC’s tire/track and spindle-coupled simulators are used primarily for conducting reliability and durability experiments. TARDEC’s platform simulators are primarily used for the conduct of soldier and hardware in-the-loop experiments.

Reliability and Durability Simulators

TARDEC’s reliability and durability motion base simulator capability consists of tire/track reconfigurable “poster” simulators which permits the installation of an entire vehicle system (light/heavy trucks, cars, and tanks weighing up to 45 tons etc.) to be mounted on a series of hydraulic actuators (Figure 2). The actuators provide disturbance inputs (generated from field or analytical simulation output data) into the vehicle system representing forces and accelerations that would be experienced in an actual terrain environment.
The second type of reliability and durability simulator (Figure 3), which is an extension of the first type, is the Pintle Motion Base Simulator (PMBS). This simulator provides the unique capability to conduct experiments on lunette trailer systems under repeatable realistic dynamic conditions, prior to or in lieu of expensive field testing. The PMBS contains electronics, software, and fixturing to provide the addition of lateral, longitudinal, and vertical forces/accelerations to a trailer’s lunette to account for the interaction dynamics between the trailer and its prime mover. Excellent correlation with field data has been demonstrated on the lunette at frequencies of up to 40 Hz and on the spindle locations at frequencies of up to 60 Hz.

The TARDEC Simulation Laboratory possesses two high bandwidth motion simulators to perform soldier and hardware-in-the-loop interactive simulation studies to examine the effects of off-road conditions on vehicle systems and subsystems as well as the interactions with the human occupants. Along with high-end computational facilities and cueing hardware, these simulators offer high fidelity, interactive, real-time simulations. The most significant component of TARDEC's physical simulation capability is its six-degrees-of-motion-freedom Crew Station/Turret Motion Base Simulator (CS/TMBS) shown in Figure 4. The CS/TMBS is a unique Stewart Platform configuration used to reproduce dynamic conditions encountered by combat vehicle crew stations and turret systems (up to 25 tons) traversing a variety of virtual terrain environments. Specific applications include the study of soldier-machine interface problems, gun turret drive weapon stabilization systems, and other issues related to the operation of turret systems and their components.

The Ride Motion Simulator is a six degree-of-freedom Stewart Platform configuration capable of recreating the ride motion, as a result of terrain dynamic conditions, of any land-based vehicle system (Figure 5). This simulator has a reconfigurable cab mounted on the platform that can support a payload of 1000 lbs. at approximately a 40 Hz positional bandwidth.

Future Laboratory Enhancements TARDEC engineers are continuously developing state-of-the-art capabilities to keep TARDEC, RDECOM, and the U.S. Army on the cutting edge of modeling and simulation.
technology. The aforementioned Suspension Characterization Test Fixture (SCTF) will become synonymous with the VIPER. While the VIPER has the capability to physically measure mass and inertia properties of a vehicle system or subsystem accurately, the SCTF will complement the VIPER and be have the capability to physically measure suspension characteristics. The current methodology of capturing these characteristics have been estimation. Once the SCTF is fully operational, TARDEC engineers will be able to capture these characteristics directly off vehicles. TARDEC engineers are also procuring an Advanced Tire Characterization machine to physically capture data from actual tires. The data obtained from this cutting-edge technology will help TARDEC engineer develop accurate Pacejka Carpet Plot Data Curves, as part of the Pacejka Magic Formula. Pacejka curves are used in tire modeling, and subsequently used in full vehicle dynamics models. TARDEC engineers model the forces that are generated by the tire as a result of the tire not following the road precisely. When the driver inputs steer into the tire a small degree, and the output is a correlating slip angle; if the driver where to depress the throttle, then the wheel would start to spin a bit; this gives a different ratio of wheel spin speed vs. ground speed, and thus gives a forward (longitudinal) force. With this advanced capability TARDEC engineers will have full confidence in their models’ accuracy from the ground up.

Another unique enhancement to the TARDEC Simulation Laboratory will be the Flat Rack Simulator (FRS). (Figure 6.) The FRS was developed in conjunction with Program Management Heavy Tactical Vehicles, Petroleum and Water Systems, and Bridging (PM-Heavy, -PAWS, -Bridging). The FRS will be used to excite motions, proper accelerations, strain levels, etc., into a FlatRack specimen on which many U.S. Army vehicles have. The FRS will be designed to impart vertical, pitch, and roll motion to the attached FlatRack (which, in this case, will serve as a test platform). These motions are designed to impart the forces that the Heavy Expanded Mobility Tactical Truck-Load Handling System/Palletized Loading System (HEMTT-LHS/PLS) would be subjected to if it were being driven over a given terrain. By correctly reproducing the vertical forces imparted to the HEMTT-LHS/PLS at the tires on the field, the simulator is capable of realistically replicating the dynamic environment experienced by traversal over a given terrain within the limits of the simulator. Along with the FRS, TARDEC engineers have software that can analyze the data recorded from the test and software, which can estimate fatigue damage given a strain time history and appropriate material properties.

Figure 6: Conceptual Drawing of the Flat Rack Simulator

Conclusion

Research Engineers and Scientists at RDECOM-TARDEC have unique computer-based analytical modeling and physical simulation laboratory tools available to support military vehicle simulation experiments. Using analytical dynamic modeling, vehicle parameter characterization, and durability/man-in-the-loop motion base simulation technologies, RDECOM-TARDEC engineers are replicating off-road dynamic behaviors of military vehicles that are realistic, repeatable, and validated. At the end of the day, the results are then used for development and performance evaluations of fielded and future military vehicle systems to assist the U.S. Army in its quest for Simulation Based Acquisition.

References


SIMULATION IN ROBOTICS
Computing with Activities IV: Chunking and Aspect Integration of Complex Situations by a New Kind of Kohonen Map with an Integrated Long-Term Memory Structure

Matthias Reuter
CUTEC Institut
Leibnizstr. 21+23
D-38678 Clausthal-Zellerfeld
matthias.reuter@cutec.de

Sabine Bostelmann
TU Clausthal
Julius-Albert-Str. 4
D-38678 Clausthal-Zellerfeld
sabineb@in.tu-clausthal.de

KEYWORDS
Learning Strategies, Computing with Activities, Whu-Structures, Long-Term Memory Structure

ABSTRACT
In this paper we present a new theory and give examples of a new kind of network structure, which shows oscillatory behaviour and codes the aspects of complex situations more exactly than it can be expected by a common neural net. It will be shown, that by introducing a long-term memory structure to SOMs, the aspect integration can be done more effectively and space saving. The long-term memory structure is able to extract known aspects of new situations and puts down only new aspects. Thus it is able to learn new aspects from complex situations without saving again all its information. In combination with the whu-structures, it completes the Chunking process, for the chunk itself is build up from the simple basic situations stored in the long-term memory structure and the whu-structures secure their connections.

THEORETICAL FRAMEWORK
Computing With Activities

As shown in (Reuter 1999) and (Reuter 2001) it makes sense to code complex situation vectors not only by the activity of one (winner) neuron but by the activity scheme of a more or less complex set of neurons. Especially if such a set is placed on a closed SOM (a closed neuron grid), one can expect that all components of a system state vector of higher dimensions (system state vector coding a large number of aspects) can contribute to a scene analysis by a neural net.

As on a closed grid every neuron has the same number of neighbours, the resulting activity pattern of the map shows symmetrical forms (as shown in Figure 1) and no information stored in the neighbourhood of a classification concept will be lost because the representative neuron of the concept is placed at the margin of the SOM.

This fact is most important for all kinds of steering and ruling systems with involved time trajectories as now a consistent continuation of a trajectory to all directions is guaranteed.

Figure 1: Classification Result of a Trained Closed SOM (3-Dim. and 2-Dim.)

In Figure 2 a complex situation categorization by the activity pattern of a closed SOM is shown, whereat the arrow marks the winner region of the last classification step.

Figure 2: Activity Structure of a Closed SOM Representing a Situation of a Soccer Match

Of course the region of the winner neuron (and the winner neuron itself) represents the scene globally but the detailed aspects of the classified scene are coded in the global activity pattern of the SOM. If the situation changes slightly, obviously the coordinates of the winner region will not change, but the oddment of the activity pattern will be modified. That means, that only if the situations – coded by the common SOM learning strategies – change, the winner region will move accordingly. If only inferior situation aspects change, the winner region keeps its position.
If the situation totally changes, the general form of the activity will change accordingly, whereby the ‘global state dynamics’ can be detected by the flux of the winner region, which is marked by the arrow in Figure 3.

![Figure 3: Activity Structure of the SOM Representing a Different Soccer Situation](image)

Based on these examples it should be clear, that the principle of the ‘Computing with Activities’ enables the continuous supervision of the real-time behaviour of a system.

**Theory Of The Whu-Structures**

Even if the method of ‘Computing with Activities’ demonstrates totally new aspects for the supervision of complex systems, it will be more convenient if sub-patterns of high relevance can be extracted from the global activity pattern. This can be done by using two different kinds of ‘post-processing strategies’. The first of them defines a bias function, which emphasizes the neurons with the largest activities. Applying the second method however, a new structure on the neuron grid has to be defined, which enables a kind of ‘situation relevant interaction’ between the neurons of interest. In our experiments we tried both methods and detected that the first one is not an adequate way to deal with ‘activities of higher interest’ as the postulated bias function has to be case-related or recalculated from classification step to classification step.

Indeed the second way to emphasize the activity pattern of interest works more convenient and - as we have to learn from neurology - is closer to the principal of how brain works. For that reason we define a new kind of ‘activity-oriented’ neighbourhood on the SOM called ‘wide hook up’-structure (resp. whu-structure). The resulting net is more or less a SOM with an arbitrary activity-oriented neighbourhood function, which simulates an involved short-time memory structure. Basic assumptions of this whu-structure are:

- The used neural structure represents a complete graph with an arbitrary neighbourhood function, which is defined by the weights of the whu-structure.
- The modified SOM contains lateral trainable connections, which can change their internural weights by a special training step.
- During every training or classification step a whu-structure is build up by those neurons that can maintain or enlarge their activity by the lateral activity transfer. That means that the whu-structure on the neuron graph forms something like a short cut or ‘conversation cycle’, which empowers the involved neurons to stabilize and/or intensify his or her own activity during a training or classification step.
- The neurons (and optionally the whu-structures) abate their activities (weights) in an exponential way over the next calculation steps. In that way a time-dependence of the short-time memory structure is defined.

**Long-Term Memory Structure**

The long-term memory structure supports the whu-structures in enabling a chunking process: it recalls simple situations that the whu-structure can use for creating more complex situations.

The long-term memory structure basically records linearly independent activity structures in an adequate 0-1-encoding. This encoding shows which neurons essentially build up the momentary activity structure. By means of this procedure it is possible to filter known aspects out of actual situations and to save the new aspects separately – thus it is not necessary to save the whole situation whereas chunking is guaranteed.

To give an example: Let $A$ be a situation, coded as a basic situation as $(1,0,1,0,0)$. Let $A'$ be the following situation, coded as $(1,1,1,0,1)$. $A'$ contains the situation $A$ as well as new aspects. According to the structure of the long-term memory structure, the known aspects of the situation $A'$ are filtered and solely the new aspects, $(0,1,0,0,1)$ in this example, are saved.

Now let there be a long-term memory structure containing some basic situations like $(1,1,0,0,0)$, $(0,0,1,0,0)$ and $(0,0,0,0,1)$. Let $(1,1,1,0,1)$ be the following situation. For this situation can be created from the already known situations, it is not necessary to save the actual situation. Thus disc space will be saved and complex situations can be reduced to known simple basic aspects.

Besides, the partitioning of complex situations in basic situations enables a substantially more flexible classification.

**EXPERIMENTAL RESULTS**
For the first experiments we chose a typical SOM enlarged by a whu-structure as shown in Figure 4. Please note that normally one calculation step of such a structure is as follows:

- First the input pattern reaches the classical SOM structure and creates a global activity pattern.
- Now all neurons transmit their evoked activity through the interneuronal connections to the other neurons on the SOM.
- Next the algorithm scans which neurons form a ‘short cut component’ that is strong enough to maintain an activity pattern on the SOM.
- If the whu-structure should be trained, these short cut weights are reinforced.

It is important to mention, that until now the aspect integration done by a whu-structure has to be trained. This training is done by modifying the classical connections of a closed SOM (input layer → SOM) in a first training phase. Then the lateral connections are installed whereas a classical Hebb learning strategy is used.

Figure 5: Whu-Structure Activity on a Closed SOM
Computing with activities this way, it is logical that activities – caused by complex situations like the classification of a robot soccer game – influence each other. In addition special situation activity presentations can be intensified or understatet by changing the interneural weights of the whu-structures. Furthermore now the underlying activity contributions of the neurons involved in a whu-structure can boost this structure so far that at the end of a calculation step ‘special activity structures’ will be more dominant than the single activities of the involved neurons. So a special detector of interesting input data constellations can be formed.

Computing with activities and whu-structures also means, that these nets show pseudo-oscillations too. That means that complex systems will remember the past by the exponential decay of the activity structures over a variable time even if the net is uncoupled from input. To examine this behaviour we designed a hyper-classificator structure of three downstream nets whereby the activity patterns of the lower nets serve as input patterns for the upper nets. If the activity of such a structure is calculated, a whu-structure is initiated like exemplary shown in Figure 6.

Figure 6: Exemplary Hyper-Classificator Structure with Activated Whu-Structures

An example of the results of our simulator is shown in Figure 7. The dimensions of the networks have been 5*5 neurons of networks number one and two and 25*25 neurons of net number three. Figure 7 shows the initial activity contributions of these three nets. Please note that the upper part of the Figure shows the landscape of the all-around activities, while the lower left hand side shows the numerical activity values of the first net. On the lower right hand side the visualized cube shows the activity pattern in that way, that the whu-structures (with the high activities) are the lines on the top of the cube while the ‘normal’ activities are placed on the bottom of the cube.

As clearly can be pointed out, at the beginning of the experiment all nets show a typical activity structure. At the next step of the experiment the upper net was uncoupled from the lower structures. Now the activity of the larger net
structure subsides very rapidly, as can be seen by the little height of the activity structure on the upper part of Figure 8. Nevertheless the principal form of the whu-structure did not change very much which is indicated by the large number of lines still located at the top of the cube shown on the lower part of Figure 8.

Figure 7: Whu-Structure Simulation

Some calculation steps later the activity structure of the hyper-classificator decreased a little bit more, but the principal form of the whu-structure still ‘survived’. It takes another seven calculation steps until the whu-structure totally vanishes.

Figure 8: Whu-Structure Simulation after the Upper Net is Disconnected

Using whu-structures also implies the application of Chunking, which is known from cognitive science. Chunking means, that so far separated concepts, (like ’apple’ and ‘tree’) will be combined to a new concept (‘apple tree’) without loosing the information of the several parts of the Chunk. Especially for creating complex neural nets for ruling and steering robots, Chunking will play a fundamental role, as it is the only way to teach machines even complex situations without running in a ‘mental overload’ resp. storing thousand of useless (similar) concepts in a net structure. Also the problem that new facts – that were hard to train at first – will be easily to handle after a while, can be understood by Chunking processes, as now it seems to be logic, that the new information from today will only be a subpart of the memory of tomorrow.

To support the chunking process and the partitioning of complex situations in simple basic situations, the long-term memory structure comes into play. An exemplary long-term memory structure is shown in Figure 9. On the lower right of the Figure the momentary activity structure is shown. This structure is converted to a pattern that will be stored in the long-term memory structure. This pattern is shown on the lower left of the Figure. Because of the very basic example, the activity structure and the pattern to be recalled by the long-term memory structure almost coincide. If the resulting pattern is independent of the already known patterns, it will simply be stored, as can be seen on the upper part of the Figure.

Figure 9: An Exemplary Long-Term Memory Structure

The next step is to disassemble complex situation in basic situations. In Figure 10 an activity structure is presented to the long-term memory structure that is partly already known. Therefore it is unnecessary to store the whole pattern again but it is important to extract unknown aspects from the complex situation. The momentary pattern – received from the converted activity structure – is again shown on the lower left of the Figure. As can be seen, it is composed of pattern 2 (second pattern from the top on the left, circled) and a new component that will be stored as pattern 6 (first pattern from the top on the right, circled).

Thus the adaption of a neural net with a long-term memory structure and a whu-structure will first handle all kinds of information at the same ‘high level’ and select the relevant (surviving) parts of the information after the evaluation of the necessary parts of the input patterns. The long-term memory structure will detect basic situations and supports the chunking process.
Figure 10: Decomposition of a More Complex Activity Structure in Two Patterns

SUMMARY

The implementation of closed Kohonen maps empowers such nets to supervise the real-time behaviour of even very complex systems by trajectories. If not only the classical winner neuron is supervised but the classification behaviour of the net is analysed in the context of the ‘Computing with Activities’, aspect integration can be done. That gives the possibility to supervise even slight system changes by the global change of the all-around activity pattern of the net. If furtheron whu-structures are implemented, the relevant aspects of an activity pattern can stabilize themselves, whereas memory structures on the map are defined. These memory structures show an oscillatory behaviour over a defined time window, whereby the width of this window can be chosen by the parameter of the exponential decay of the activity of the whu-structure. The long-term memory structure remembers basic aspects and is a helpful tool to create complex situations out of basic situations.

REFERENCES


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 University of Clausthal. He developed a couple of new neural nets structures and learning strategies. His major investigations areas are neuronal net dynamics and robotics.

SABINE BOSTELMANN was born in Hannover, Germany and went to the University of Hildesheim and to the Technical University of Clausthal, where she studied business mathematics. She worked at the University of Dortmund and at the Technical University of Clausthal. Her favourite subject of research are neural dynamics and the evaluation of new learning strategies of neural nets.
GMDH TYPE NEURAL NETWORKS AND THEIR APPLICATION TO THE IDENTIFICATION OF THE INVERSE KINEMATIC EQUATIONS OF ROBOTIC MANIPULATORS

Ahmad Bagheri
Nader Nariman-Zadeh
Ardavan Sufi Siavash
Alireza Khoobkar
Department of Mechanical Engineering, Guilan University
P.O.Box 3756, Rasht, IRAN
E-mail: bagheri@gilan.ac.ir

KEYWORDS
GMDH, Neural Networks, Inverse Kinematics, Puma760.

ABSTRACT

In this paper, it is shown that group method of data handling (GMDH)-type neural networks and their application can be effectively used to acquire the inverse kinematic equations of a Puma760 robot manipulator based on the numerical data of its motion. The aim of such modeling is to show the accuracy of GMDH-type neural networks. For evaluating the accuracy of the obtained equations, a new trajectory is employed to demonstrate whether the models are still valid or not. Finally, the best results are used to define the inverse kinematic equations.

INTRODUCTION

The inverse kinematic equations of robotic manipulators are used to identify the angular positions of the links, when the Cartesian coordinate of end-effector is known (Craig 1989). Since identification and modeling of such equations using input-output data is one of the time-consuming and complicated problems in the control of robotic manipulators, several system identification techniques are applied to model the inverse kinematics of robotic manipulators.

Neural networks, which is one of the main components of soft-computing and have shown great ability in solving complex non-linear system identification and control problems, could be effectively used for this purpose. Among several methodologies that have been expanded, group method of data handling (GMDH) algorithm is a self-organizing approach by which gradually complicated models are generated based on the evaluation of their performances on a set of multi-input single-output data pairs. In this way, GMDH was used to overcome the difficulty of knowing a priori knowledge of mathematical model of the system. The main idea of GMDH is to build an analytical function in a feedforward network based on a quadratic node transfer function whose coefficients are obtained using regression technique.

KINEMATICS OF A PUMA760 ROBOT

PUMA760 is a spatial 6 DOF robotic manipulator with all revolute joints (Fu et al. 1990). A schematic diagram of this robot is shown in Figure 1.

![Figure 1: A Schematic Diagram of a Puma760 Robot](image)

In order to obtain the kinematic equations, the Denavit-Hartenberg parameters of this robot are shown in Table 1. Using such parameters, appropriate transfer matrices of links and also the kinematic equations can be found.
Table 1: The Denavit-Hartenberg Parameters

<table>
<thead>
<tr>
<th>Joint i</th>
<th>$\theta_i$</th>
<th>$\alpha_i$</th>
<th>$a_i$</th>
<th>$d_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>90</td>
<td>-90</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>700</td>
<td>300</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>90</td>
<td>-33</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>-90</td>
<td>0</td>
<td>915.1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>90</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>170</td>
</tr>
</tbody>
</table>

The considered trajectory of a Puma760 is shown in Figure 2.

![Figure 2: The Considered Trajectory](image)

**USING GMDH TYPE NEURAL NETWORKS IN MODELING**

The classical GMDH algorithm can be represented as a set of neurons in which different pairs of them in each layer are connected through a quadratic polynomial and thus produce new neurons in the next layer. Such representation can be used to model the mapping inputs to outputs. The formal definition of the identification problem is to find a function $\hat{f}$ so that can be approximately used instead of actual one, $f$, in order to predict output $\hat{y}$ for a given input vector $X=(x_1,x_2,x_3,...,x_n)$ as close as possible to its actual output $y$. Therefore, given $M$ samples of multi-input single-output data pairs define the following equations:

$$y_i = f(x_{i1},x_{i2},x_{i3},...,x_{in}), \quad i = 1, 2, ..., M. \quad (1)$$

It is now possible to train a GMDH-type neural network to predict the output values $\hat{y}_i$, for any given input vector $X=(x_{1i},x_{i1},x_{i2},x_{i3},...,x_{in})$. It means,

$$\hat{y}_i = \hat{f}(x_{1i},x_{i1},x_{i2},x_{i3},...,x_{in}), \quad i = 1, 2, ..., M. \quad (2)$$

The problem is now to determine a GMDH-type neural network so that the square of difference between the actual output and the predicted one to be minimized,

$$\frac{1}{M} \sum_{k=1}^{M} \left[ (\hat{y}_i - y_i)^2 \right] \rightarrow Min. \quad (3)$$

General connection between inputs and output variables can be expressed by a complicated polynomial of the form

$$\hat{y} = a_0 + \sum_{i=1}^{m} a_i x_i + \sum_{i=1}^{m} a_{ij} x_i x_j + \sum_{i=1}^{m} a_{ijk} x_i x_j x_k + ... \quad (4)$$

which is known as the Ivakhnenko polynomial (Farlow 1984). However, for most applications, the quadratic form of only two variables is used in the form of

$$\hat{y} = G(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 x_i^2 + a_4 x_j^2 + a_{5i} x_i x_j \quad (5)$$

to predict the output $y$. The coefficients $a_i$ in equation (5) are calculated using regression techniques (Ivakhnenko 1981; Iba et al. 1986) so that the difference between actual output, $y$, and the calculated one, $\hat{y}$, for each pair of $(x_i, x_j)$ as input variables to be minimized. Indeed, it can be seen that a tree of polynomials is constructed using the quadratic form given in equation (5), whose coefficients are obtained in a least-squares sense. In this way, the coefficients of each quadratic function $G_i$ are obtained to optimally fit the output in the whole set of input-output data pair,

$$r^2 = \frac{\sum_{i=1}^{M} (y_i - G_i(x_i))^2}{\sum_{i=1}^{M} y_i^2} \rightarrow Min. \quad (6)$$

In basic form of the GMDH algorithm, all the possibilities of two independent variables out of total $n$ input variables are taken in order to construct the regression polynomial in the form of equation (5) that best fits the dependent samples $(y_i, i = 1, 2, ..., M)$ in a least-squares sense. Consequently, $\binom{n}{2} = n(n-1)/2$ neurons will be built up in the second layer of the feedforward network from the samples $\{(y_i, x_i, x_{iq}), (i = 1, 2, ..., M)\}$ for different $p,q \in \{1, 2, ..., M\}$ (Farlow 1984). In other words, it is now possible to construct $M$ data triples $\{(y_i, x_{ip}, x_{iq}), (i = 1, 2, ..., M)\}$ from samples using such $p,q \in \{1, 2, ..., M\}$ in the following form:

$$\begin{bmatrix}
  x_{1p} & x_{1q} & : & y_1 \\
  x_{2p} & x_{2q} & : & y_2 \\
  \vdots & \vdots & \vdots & \vdots \\
  x_{Mp} & x_{Mq} & : & y_M
\end{bmatrix}$$

Using the quadratic sub-expression in the form of equation (5) for each row of $M$ data triples, the following matrix equation can be readily obtained as

$$A a = Y \quad (7)$$
where \( a \) is the vector of unknown coefficients of the quadratic polynomial in equation (5):

\[
a = \{a_1, a_2, a_3, a_4, a_5, a_6\}
\]

and

\[
Y = \{y_1, y_2, y_3, \ldots, y_N\}^T
\]

is the vector of output values from samples. It can also be readily seen that

\[
A = \begin{bmatrix}
1 & x_{1p} & x_{1q} & x_{1p}^2 & x_{1q}^2 \\
1 & x_{2p} & x_{2q} & x_{2p}^2 & x_{2q}^2 \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
1 & x_{Mp} & x_{Mq} & x_{Mp}^2 & x_{Mq}^2 \\
\end{bmatrix}
\]

The least-squares technique from multiple-regression analysis leads to the solution of the normal equations in the form of

\[
a = (A^T A)^{-1} A^T Y
\]

which determines the vector of the best coefficients of the quadratic equation (5) for the whole set of \( M \) data triples. However, such solution directly from normal equations is rather susceptible to round off error and, more importantly, to the singularity of these equations.

**STRUCTURAL IDENTIFICATION OF GMDH TYPE NEURAL NETWORKS**

Three different approaches for structural identification of GMDH-type networks are presented as follows: (Nariman-Zadeh et al. 2002).

**Method I: Increasing Selection Pressure Approach (ISP)**

In this approach, only one parameter, called selection pressure, is sequentially increased in different layers in order to determine the number of neurons in each layer and also the number of layers in network. The main steps of this approach are described as follows:

**Method II: Pre-specified Structural Design Approach (PSD)**

In this approach, the number of layers in the network and also the number of neurons in each layer is pre-specified.

**Method III: Error Driven Structural Approach (EDS)**

In this approach, the numbers of layers as well as the number of neurons in each layer are determined according to a threshold error for equation (6). In addition, unlike two previous approaches, some of input variables or generated neurons in different layers can be included in subsequent layers. It is, therefore, evident that the structure of such network may be more complicated than those generated in previous methods.

**GMDH TYPE NEURAL NETWORKS MODELING OF THE INVERSE KINEMATIC EQUATIONS OF A PUMA760 ROBOT**

Three methods discussed previously are used to design GMDH-type network systems for a set of actual input–output data in a series of Puma760 trajectory obtained from Roboworks (www.newtonium.com).

Selected parameters of interest in this multi-input six-output system, which affect the main and local coordinates of gripper, are \( X, Y, Z, T, A, O \). The \( X \), \( Y \) and \( Z \) are defined as the hand coordinates of the gripper and also \( T \), \( A \) and \( O \) are defined as the Tool, Altitude and Orientation angles of the gripper respectively (Fu et al. 1987).

The complete data set consists of a total number of 25 input–output actual data considering different value of inputs and six output parameters. In order to model these six-input six-output set of data, each of the three methods previously mentioned was used separately in conjunction with singular value decomposition (SVD) approach for the coefficient of the quadratic polynomials. The actual data obtained from the Roboworks constitute six sets of 25 six-input single-output data used independently by these three methods.

Figures 3-8 show the modeling behaviour of identified networks. Accordingly, figures 9-14 show the structure of identified networks.

**Figure 3: Variation of joint 1 angle with input data samples:**

Comparison of actual values with computed values by method II (PSD) – 2 layers

**Figure 4: Variation of joint 2 angle with input data samples:**

Comparison of actual values with computed values by method I (ISP) – 2 layers
Figure 5: Variation of joint 3 angle with input data samples: comparison of experimental values with computed values by method I (ISP) – 2 layers

Figure 6: Variation of joint 4 angle with input data samples: comparison of experimental values with computed values by method III (EDS) – 4 layers

Figure 7: Variation of joint 5 angle with input data samples: comparison of experimental values with computed values by method III (EDS) – 4 layers

Figure 8: Variation of joint 6 angle with input data samples: comparison of experimental values with computed values by method III (EDS) – 4 layers

Figure 9: GMDH-type network obtained by method II (PSD) for Joint 1

Figure 10: GMDH-type network obtained by method I (ISP) for Joint 2

Figure 11: GMDH-type network obtained by method I (ISP) for Joint 3
In this way, Figures 16-21 show the validation of the obtained GMDH-type networks by contrast of three different approaches. The best obtained network for each joint is used to define the inverse kinematic equations which are indicated in Equations 12-17.

By comparison of graphs in three different approaches, it is evident that the PSD method reveals the best result. Therefore, the inverse kinematic equations of joint 1 are concluded as follows:

\[ Y = 0.0003 + 0.0834H_1 + 0.9332H_3 + 16.5379H_4, H_1^2 - 8.1296H_2^2 - 8.4116H_3^2 \]
\[ H_1 = 0.4989 - 0.0124X - 0.02864 + 0.0003X^2 + 0.5582X^2 \]
\[ H_2 = 4.9610 + 0.0097Z + 32.3643O - 0.2847Z + 0.0001Z^2 + 0.9346O^2 \]

By comparison of graphs in three different approaches, it is evident that the both ISP and PSD methods reveal the best result. Therefore, the inverse kinematic equations of joint 2 are shown as follows:

\[ Y = -0.1817 + 4.9810H_1, -3.5545H_3, 0.6043H_4, -0.0738H_5^2 + 0.6124H_5^2 \]
\[ H_1 = 68.1428 - 3.0333Y - 0.4215Z - 0.8853YZ + 0.0203Y^2 + 0.0024Y^2 \]
\[ H_2 = 15.7548 - 0.4215 Y + 57.4692O - 1.7071YO + 0.0046Y^2 + 2.3713O^2 \]
By comparison of graphs in three different approaches, it is evident that the ISP method reveals the best result. Therefore, the inverse kinematic equations of joint 3 are shown as follows:

\[ \begin{align*}
Y &= 1.9209 + 16.2267H_1 + 15.871H_2 + 1.1267H_3 + 1.8225H_4 + 0.7496H_5^2 \\
H_1 &= 5.0068 + 0.027X - 0.027X + 0.0330X + 0.0001X^2 + 0.5252O^2 \\
H_2 &= 5.1970 - 0.0018T - 1.2276O - 0.0015TO + 0.0001T^2 - 0.3842O^2
\end{align*} \]  

By comparison of graphs in three different approaches, it is evident that the EDS method reveals the best result. Therefore, the inverse kinematic equations of joint 4 are shown as follows:

\[ \begin{align*}
Y &= 0.0004 - 0.0762H_2 + 1.0718H_1 - 0.2576H_1H_2 + 137.6804H_2^2 + 137.3714H_2^3 \\
H_1 &= 0.0528 - 0.0006Z + 1.0180H_2 + 0.0003ZH_2 + 0.0097H_3^3 \\
H_2 &= 0.5430 - 0.0166X - 0.2149H_3 + 0.0108XH_3 + 0.0001X^2 + 0.3953H_3^2 \\
H_3 &= 1.5364 + 0.0076Z + 7.5824O + 0.0582Z + 0.3991O^2
\end{align*} \]  

By comparison of graphs in three different approaches, it is evident that the EDS method reveals the best result. Therefore, the inverse kinematic equations of joint 5 are shown as follows:

\[ \begin{align*}
Y &= 0.0697 - 0.0164H_1 + 0.9721H_1 + 0.0096AH_1 - 0.0031H_2^2 + 0.0028A^2 \\
H_3 &= -0.3607 - 0.0109Z + 1.7327H_2 + 0.0012ZH_2 - 0.1296H_2^2 \\
H_2 &= -1.2548 - 0.0056Z + 2.4H_1 + 0.0002XH_1 - 0.2260H_1^2 \\
H_1 &= 5.8117 - 0.0048Z + 0.0004O - 0.0001Z^2 - 0.4272O^2
\end{align*} \]  

By comparison of graphs in three different approaches, it is evident that the EDS method reveals the best result. Therefore, the inverse kinematic equations of joint 6 are shown as follows:

\[ \begin{align*}
Y &= 0.9300 + 0.0044H_3 + 0.5690Y - 0.0010YH_1 + 0.0490Y^2 \\
H_3 &= -0.8661 + 0.0231Y + 1.1749H_2 - 0.0025YH_1 - 0.0001Y^2 - 0.0082H_1^2 \\
H_2 &= 1.9655 + 0.0113Z + 0.0782H_1 - 0.0027ZH_1 - 0.1063H_1^2 \\
H_1 &= 6.0544 + 0.0108X + 0.0052A - 0.0001AX - 0.0001X^2 - 0.2331A^2
\end{align*} \]  

It can be seen clearly that the performance of method III in the GMDH-type neural network modeling of the inverse kinematic equations in most cases is superior to those of both methods I and II.
CONCLUSION

The effectiveness of GMDH-type of neural networks which could model the complex systems without having specific knowledge of the systems, is shown in this paper. The results presented in this paper clarified that GMDH-type networks can precisely model the inverse kinematic equations of a Puma760 robot manipulator. In addition, it is clear that this approach can be used for identifying the inverse kinematic equations of the n DOF robotic manipulators. Moreover, it has been shown that SVD can effectively improve the accuracy of such GMDH-type networks which can be constructed by each of the three methods.

REFERENCES


[URL]: www.newtonium.com
SIMULATION
IN
ELECTRONICS
ELECTRONICS PRODUCTION SIMULATION
EVALUATION OF WORKPIECE ROUNDNESS ERROR IN A TURNING OPERATION: AN ASSESSMENT OF COMPUTATIONAL ALGORITHMS BY USING SIMULATED DATA

Massimo PACELLA*, Alfredo ELIA and Alfredo ANGLANI
Dipartimento di Ingegneria dell’Innovazione
Università degli Studi di Lecce
Via per Monteroni, 73100 Lecce, Italy

KEYWORDS
Probabilistic model; Spectral analysis; Manufacturing quality control.

ABSTRACT
Circular or cylindrical features are fundamental geometric features in quality control of manufacturing products. At present, the standard for geometric dimensioning and tolerancing specifies a circularity tolerance based on the tolerance zones defined by two concentric circular boundaries. The purpose of our research is to investigate the effect of measurement error resulting on the form tolerance estimation, obtained by some form fitting algorithms. This paper presents a study for roundness tolerance modelling and analysis, as well as a software tool for assessing the implemented fitting algorithms. To represent profile variations within the tolerance zones, a simulation tool, which is based on a harmonic roundness model using Fourier series expansions, has been used. The simulation tool has been implemented within the Matlab environment and includes both the least square and minimum zone tolerance fitting algorithms.

INTRODUCTION

A coordinate measuring machine (CMM) is a computer-controlled device for assessment of manufactured products. CMMs are widely used in the inspection of the geometry and dimensions of mechanical parts for statistical quality control and process inspection of manufacturing systems. Through a variety of mechanisms, these machines determine the position of points on the surface of the object under inspection. After a set of point measurements has been collected, part conformance to the design target can be evaluated. The operation of CMMs consists of two phases: 1) measurements of coordinates of points on the machined surface, and 2) estimation of the geometric features of the workpiece, using sampling data and some form fitting algorithms.

In the second phase, in order to produce the desired results for inspection, data obtained from CMMs are analyzed using appropriate computer-based algorithms, and these algorithms must follow specifications laid down in standards. Moreover, these algorithms must be efficient, robust, and should consume optimal time for producing the results. Although CMMs are characterized by high measuring precision, there are still inaccuracies and uncertainties associated with them. Sources of these errors are: 1) measurement machine components and probes, 2) sampling strategy, 3) data processing, 4) measurement environment. For many CMM users, it would be useful to know effects of measurement errors on estimation results obtained by using specific fitting algorithms. However, at present, little is known about the sensitivity of data processing algorithms to measurement errors.

The purpose of our research is to investigate the effect of CMM measurement error resulting on the form tolerance estimation, obtained by algorithms commonly implemented in CMMs. In this study, effects of CMM measurement error on form tolerance estimation have been investigated through computer simulation of a basic feature (the circle resulting from a turning operation), by using standard CMM tolerance evaluation algorithms: the least square (LS) and the minimum zone (MZ).

To such an aim, a software tool has been implemented within the Matlab environment. A schema of the implemented system has been depicted in Figure 1. It consists of two main parts: 1) a profile simulator and 2) a profile verification system. The implemented simulation environment allows users to simulate turning profiles as well as to use actual measurement data points.

![Figure 1 – Software Main Scheme](diagram)

This paper is structured as follows. Sections 2 and 3 describe the model for generating simulated profiles and verification algorithms used to perform fitting analyses. Section 4 describes a probabilistic model based on the harmonic spectra. Subsequent sections 5-10 describe fitting and verification algorithms. Finally a test case and some concluding remarks are discussed.

A HARMONIC MODEL OF ROUNDNESS

Generally, measurement data obtained from inspection machines such as CMMs are in Cartesian coordinates. Moreover, typical diameters of components produced by

* Corresponding author, fax: (+39) 0832 297 279
e-mail: massimo.pacella@unile.it
turning operations are of the order of millimetres, and circularity deviations are in microns. The purpose of CMM data fitting is to apply an appropriate algorithm on sample data points obtained from the part being inspected, in order to fit a geometric form. The perfect form approximation obtained through fitting is called a substitute feature.

The ANSI Dimensioning and Tolerance Standard Y14.5 specifies that form tolerances on a component must be evaluated with reference to an ideal geometric feature (ASME Y14.5M). ISO standard recommends to use minimum zone algorithm to evaluate the ideal/reference feature, so that deviation from it and the actual feature concerned will be as least as possible (ISO 1101-1983). However, neither of these standards specifies methods for establishing the ideal feature and evaluation of form errors.

For a circular profile resulting from a turning operation, ISO standard for geometric dimensioning and tolerancing specifies a circularity tolerance based on the tolerance zones defined by two concentric circular boundaries to confine both the highest peak and the lowest valley of a roundness profile. The variations of a profile within extreme boundaries are not defined, thus, parts satisfying the same circularity tolerance can have very different profiles.

Various researchers have attempted to develop methods for establishing the reference feature and to evaluate circularity error. Cho and Tu (2001) proposed a harmonic roundness model using Fourier series expansion to describe profile variations within the two confining circular boundaries. The use of harmonics is justified because machined circular profiles tend to be periodic due to rotational error motions of machine tools. A harmonic roundness model assumes a circular profile as a series of harmonics superimposed on an ideal circle. By using Fourier series expansions, the harmonic roundness model can be written as:

\[ r(\theta) = R + \sum_{k=1}^{N} \left[ A_k \cos(k\theta) + B_k \sin(k\theta) \right] \]

(1)

\[ C_k = (A_k^2 + B_k^2)^{1/2} \]

(2)

\[ \Phi_k = \tan^{-1}(A_k / B_k) \]

(3)

In equation (1) \( C_k \) is the amplitude of the \( k \)th harmonic, while \( \Phi_k \) is its phase angle. Ideally, an infinite number of harmonics and data points is needed to completely reconstruct a machined circular profile, which is impossible in practice; however, most of the information it has been demonstrated to be contained in the first 50 components.

For a given profile \( r(\theta) \), composed of \( N \) discrete points, coefficients \( A_k \) and \( B_k \) can be estimated as follows:

\[ \hat{A}_k = \frac{2}{N} \sum_{i=1}^{N} r(\theta_i) \cos(k\theta_i) \]

(4)

\[ \hat{B}_k = \frac{2}{N} \sum_{i=1}^{N} r(\theta_i) \sin(k\theta_i) \]

Values of \( C_k \) and \( \Phi_k \) can be computed as indicated by equations (2) and (3). In reverse, the circular profile can be estimated by means of the rebuilt harmonic spectra:

\[ r(\theta) = \hat{A}_k \cos(k\theta) + \hat{B}_k \sin(k\theta) \]

(5)

\[ \hat{A}_k = \frac{1}{N} \sum_{i=1}^{N} r(\theta_i) \]

(6)

Equation (4) may be applied, once known the dataset of measure points on a real CMM profile. Following section describes how to estimate a turned profile dataset by using a statistical model based on harmonic spectra.

**STATISTICAL MODEL OF CMM PROFILES**

A statistical approach has been used in this work in order to build roundness models for parts machined by turning processes. In particular, specific statistical distribution functions for harmonic coefficients have been exploited. Cho and Tu (2001) demonstrated that the best fitting distribution function for amplitudes \( C_i \) is the beta distribution, while phase angles, \( \Phi_i \), can be assumed uniformly distributed within 0 and 2\( \pi \).

Furthermore, it has been demonstrated (Capello and Semeraro 2001a, 2001b) that the radius of fitted substitute geometry, obtained by using the least squares method (LSM), can be approximated by the amplitude of the harmonic of zero order and that the position of the centre can be approximated by the value of the first harmonic amplitude and phase:

\[ C_0 = \hat{r}_{LSM} \]

(7)

\[ C_1 = e \]

(8)

Thus, the first order harmonic represents the spindle error motion, while the others describe the form error. In the model implemented in our research, we assume \( \Phi_i \) to be uniformly distributed within [0,2\( \pi \]), while \( C_1 \) is extracted from a normal distribution, with mean and standard deviation left user defined. It should be noticed that the harmonic amplitudes, which appear in the model proposed by Cho and Tu (2001), are normalized in order to obtain a tolerance value ranging in [0,1]. Hence, in the profile model implemented in our research, a tolerance multiplying factor for harmonic amplitudes (say \( t_i \)) from 2\( \text{nd} \) to 50\( \text{th} \) order has been introduced in order to obtain a tolerance value ranging in [0,\( t_i \].

In order to simulate different CMM measures, a phase angle \( \theta_i \), which varies from every measure, is added to \( k \)th harmonic argument \( \theta \). The phase angle \( \theta_i \) is a random variable uniformly distributed within [0,2\( \pi \]), \( \theta_i \) represents the starting angle of the CMM measure, with respect to the horizontal line.

Hence, the simulation model has been implemented as follows:

\[ r_{\text{sim}}(\theta) = C_0 + C_i \left[ \cos((\theta + \theta_{i\text{sim}}) - \Phi_i) \right] + \sum_{i=2}^{50} \frac{1}{2} C_i \cos\left( k(\theta + \theta_{i\text{sim}}) - \Phi_i \right) \]

(9)

\[ \theta_i = \frac{2\pi i}{N} \quad i = 1,...,N \]

Where \( H \) is the number of harmonics \( (H=50) \) and \( N \) is the points number on a single profile.

Equation (9) represents a simulated profile for the \( m^{\text{th}} \) measure on the \( s^{\text{th}} \) sample. Amplitudes, \( C_i \), and phase angles, \( \Phi_i \), are referred to \( k^{\text{th}} \) harmonic and \( s^{\text{th}} \) sample while phase angles \( \theta_{i\text{sim}} \) are referred to \( s^{\text{th}} \) sample and \( m^{\text{th}} \) measure, \( t_i \) is the tolerance factor for \( s^{\text{th}} \) sample while the \( \theta_i \) angles are uniformly distributed within [0,2\( \pi \)].
REBUILDING MODEL IN FREQUENCY DOMAIN

Once a real machined profile has been sampled by using an inspection machine, the roundness error can be evaluated by means of an innovative approach based on the rebuilding of the profile in the frequency domain. (Moroni et al. 2003).

Since the form error, which represents the noise around the average circular profile, depends on harmonics of order higher than one, the difference between the maximum and the minimum value of the noise may be calculated as estimate of the roundness error $e$:

$$e = \max \left( \sum_{k=1}^{H} A_k \cos(k\theta) + \sum_{k=1}^{H} B_k \sin(k\theta) \right) + \min \left( \sum_{k=1}^{H} A_k \cos(k\theta) + \sum_{k=1}^{H} B_k \sin(k\theta) \right)$$

The number of measurement points needed to evaluate equation (4) may be determined by means of the well known sampling theorem.

The sampling theorem states that it is possible to completely describe a signal with $N$ harmonics only if the sampling frequency is higher than or equal to $2N$. If the signal is sampled at a lower frequency, the phenomenon of aliasing takes place. Therefore, if a circular profile can be described by $H$ harmonics, as supposed in equation (9), the number of sampled points $N$ for a rebuilt profile should be higher than $2H$. Estimated rebuilt profile (5) obtained with $N \geq 2H$ is identical and is the best representation of the real profile. When the number of sampling points is less than $2N$, the problem of aliasing occurs and the sampled spectrum differs from the real one.

As depicted by the schema of Figure 2, the simulation environment implemented in our research allows the user to carry out the rebuilding of the simulated profile in the frequency domain.

The profile database which results from the generation algorithms consists of two multidimensional arrays $r(i,m,s,\theta(i,m,s),\gamma(i,m,s))$, representing profiles datapoints in polar coordinates, where dimensions $s, m$ and $i$ represent sample, measure and data point numbers respectively.

The frequency rebuilding model can be applied both on real measures of turning machined specimens and on simulated measures by using the harmonic model described in the previous two sections.

VERIFICATION ALGORITHMS

For tolerated features and form tolerance specifications, a domain of possible tolerance zones can be described (Carr and Ferreira 1995a). The zone is located and oriented in space by the zone locating position vector $L$ (a vector from the system reference origin to a point inside the zone that controls the location of the zone in space) and the zone direction vector $T$ (unit vector that controls the orientation of the zone in space). When the zone shape is defined by two concentric circles lying in the same plane, the direction vector $T$ is the direction normal to that plane and the locating position vector $L$ is the centre of the circles. Let $t$ be the zone size (radial distance between the two circles), and $r$ be the radial distance from the system reference origin to the reference circle of the tolerance zone, then the zone is defined by the set of points $\{P_i\}$ where:

$$|r(\theta) - r| < t/2 \quad \forall P_i \in \{P\}$$

The point-set mathematical definition of tolerance zones can be incorporated into verification algorithms to compute the minimum form error and verify the tolerance specification. The common searching methods that attempt to fit perfect-form geometry to the datapoints are based on the $L_p$-norm solution.

The $L_p$-norm solution minimizes the following function:

$$L_p = \sum_{i=1}^{n} |e_i|^p$$

where $n = \text{number of datapoints}; e_i = \text{distance (residual error) between the } i^{th} \text{ datapoint and the ideal feature}; 1 \leq p \leq \infty$. The second norm or $L_2$-norm solution (also called the Gaussian solution), obtained using least-squares algorithms, is commonly used in CMM software. This solution minimizes the sum of squares of the residual errors:

$$\min \sum |e_i|^2$$

The other solution receiving attention is the $L_\infty$ solution. This solution (also called the Chebyshev or minimum zone solution) minimizes the maximum error $|e_i|$ on both sides of the feature and results in a $\min \{|Max|e_i|\}$ objective function.

The $L_\infty$-norm problem can be transformed into a nonlinear constrained optimization problem. The difference between the maximum of a function and the minimum of the same function, and does not include the residual error term $c_i$. Only the $L_\infty$-norm solution provides a minimum zone solution. The proposed software provides an implementation of the least squares algorithm, and two different formulations for the minimum zone algorithm, as shown in Figure 3. The next sections describe each of these fitting algorithms.
LEAST SQUARES ALGORITHM

The computational aim of most assessment software used in coordinate metrology can be formulated in terms of an optimisation problem. The least-squares algorithm (LS) is the most commonly used algorithm for form tolerance evaluation. This is due both to the uniqueness of the least squares solutions and to the algorithm’s simplicity. LS method is based on sound mathematical principles that minimize the sum of squared deviations of measured points from the fitted feature. This method is robust, but it does not follow the standards intently and will not guarantee the minimum zone solution specified in the standards. In addition, deviation values and geometric tolerances that are determined by LS will be generally larger than the actual ones, and this may lead to rejection of good parts.

A circle can be described by its centre coordinates \((x_0, y_0)\) and its radius \(r_0\). The distance \(d\) from a point \(P_i = (x_i, y_i)\) to a circle defined by parameters:

\[
a = (x_0, y_0, r_0)
\]

\[
d = d(P, a) = \sqrt{(x - x_0)^2 + (y - y_0)^2 - r_0^2}
\]

Given a set of \(N\) data points \(\{P\}\), the Gaussian (least squares) associated circle is found by solving the optimization problem:

\[
\min_a \sum_{i=1}^{N} d^2(P_i, a)
\]

In the proposed software this problem is solved by using LSGE library for Matlab, developed by NPL (the UK’s National Measurement Laboratory). The Least Squares Geometric Elements (LSGE) library consists of functions to find the least-squares fit of geometric shapes to data, implementing a number of geometric fitting routines key functions. It is based on a general purpose non-linear least-squares solver that takes as input function-and-gradient routines, and these routines are implementations of the geometric evaluation key functions.

MINIMUM ZONE ALGORITHM – MINMAX FORMULATION

The minimum zone (MZ) circles are obtained when two concentric circles contain all the points of the dataset and have minimum radial distance between them. To meet these criteria, minimum zone circles should pass through at least four points of the dataset. This can occur when three points lie on one circle and the fourth point on the other circle concentric to it, or when two points are on each circle.

The first roundness verification model investigates the \(L_{c, \tau}\) norm solution that directly solves \(\min \{\max \{e_i\}\}\). This method searches for a \(T\) and \(L\) so that the distance of each datapoint from a reference cylinder is minimized.

For given roundness tolerance \(\tau\) and set of datapoints \(\{P\}\) determine values for the variables \((L_0, L, r)\) that satisfy:

\[
|r(\theta) - r| \leq \tau \quad i = 1, \ldots, N
\]

\[
\text{check : } v \leq t / 2
\]

This formulation of roundness problem results in a nonlinear discontinuous objective function solved by using the Optimization Toolbox of Matlab.

MINIMUM ZONE ALGORITHM – LINEARIZED FORMULATION

Carr and Ferreira (1995b) formulate roundness and cylindricity as non-linear optimization problems, which are then transformed into a series of linear problems. The efficiency of these algorithms with respect to computation time is reported as a linear function of the number of data points.

In this section a formulation for minimum zone solution is presented, which has been derived from the solution given by Carr and Ferreira (1995b) for the roundness problem. The \(\min\max\) formulation can be transformed into an equivalent optimization problem that has a continuous objective function:

\[
\min \{r_i - r_j\} = v
\]

\[
s.t. \\
|r(\theta) - r_i| \leq r_i, \quad i = 1, \ldots, n
\]

\[
|r(\theta) - r_j| \leq r_j, \quad r_i, r_j \geq 0
\]

\[
\text{check : } v \leq t
\]

Where:

- \(r_i\) = radial distance from axis to inner cylinder
- \(r_j\) = radial distance from axis to outer cylinder
- \(n\) = number of points in \(\{P\}\)

An algorithm based on this new formulation, which is based on the strategy of solving a sequence of constrained linear programs, has been used. Each new linear program is created from the results of the previous one and computes the next incremental change in \(L_{\min}\) required for finding the point of a local minimum solution. The non-linear constraints can be linearized by using incremental variables.

The incremental rigid body translation of \(L_{\min}\) position vector by small distances \(\Delta x, \Delta y\) is given by:

\[
L = L_{\min} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}
\]

The two variables \(r_1\) and \(r_2\) are also incrementally changed:

\[
r_1 = r_{\min} + \Delta r_1
\]

\[
r_2 = r_{\min} + \Delta r_2
\]

To improve the efficiency and robustness of the algorithm, each linear program is solved in a coordinate system that allows for a simplification of the above linear program. In particular, the problem coordinate system is translated before each linear program so that \(L_{\min}\) coincides with the origin. In this preferred coordinate system, the linearized formulation is simplified as shown below:

\[
\min \{\Delta r_i - \Delta r_j\} = v
\]

\[
s.t. \\
-2P_x \Delta x - 2P_y \Delta y - 2r_{\min} \Delta r_i \geq r_{\min}^2 - P_x^2 - P_y^2, \\
-2P_x \Delta x - 2P_y \Delta y - 2r_{\min} \Delta r_j \leq r_{\min}^2 - P_x^2 - P_y^2
\]

\[
i = 1, \ldots, n
\]

\[
-\lim_{\text{linear}} \leq \Delta x, \Delta y \leq \lim_{\text{linear}}
\]

\[
-\infty \leq \Delta r_i, \Delta r_j \leq \infty
\]

The main loop begins by determining the nominal rigid body translation of the input vector \(L_{\min}\) to the origin. The nominal rigid body translation is applied to all datapoints in \(\{P\}\), resulting in a transformed problem in a preferred coordinate system. In this system the linear program can be solved to determine incremental changes in \(L_{\min}\) that results in an improved roundness origin. Next, the roundness origin is updated by applying computed incremental changes and
transforming the problem back to the original coordinate system.

**TESTING SOFTWARE**

CMM fitting software plays an important part of computer-aided inspection in manufacturing and, consequently, it is essential that the numerical correctness of such software is verified.

The standard ISO 10360-6:2001 defines a method for testing software used to compute Gaussian (LS) associated features from coordinate measurements. Some of the geometric features covered are the line (in two and three dimensions), the plane and the circle (in two and three dimensions). The standard proposes the use of either reference software or data generator software, but is not concerned with how such software might be implemented. To such an aim, Cox et al. (2001) proposed two methods for generating reference data sets and corresponding reference results for testing Gaussian (LS) and Chebyshev (MZ) feature fitting software. The procedure proposed by the authors allows many data sets with the same solution to be generated easily, and permits data sets with specified properties to be produced.

The software implemented in our work has been successfully tested on data sets for testing both Gaussian and Chebyshev form assessment software. The general approach can be summarised in three phases: (1) determine reference data sets (appropriate for the computational aim) and corresponding reference results; (2) apply the software under test to reference data sets to produce test results; (3) compare test results with reference results.

**GUI**

Figure 4 shows an example of how the Graphical User Interface (GUI) of proposed software tool should look. It can be noticed that the GUI recalls the same structure of the logical framework depicted in Figure 1. The software tool offers two ways for analyzing profile data base: single profile mode, and all profiles mode.

An example of a graphical representation of results is depicted in Figure 6.

![Figure 5 – Single Profile Plot Example](image)

**TEST CASE**

A test case is conducted to investigate the differences between the performances of fitting methods in the evaluation of radius and form tolerance of circular profiles with an increasing degree of shape deformation, from the undeformed circular shape to an ellipsoidal shape. The deformation is simulated by increasing $C_1$, amplitude of first order harmonic. The other simulation parameters are the following:

- nominal radius = 25400 $\mu$m
- tolerance factor = 1% nominal radius
- $C_1$ = 1.5-10-15 % nominal radius
- num. of measures = 20
- num. of data points = 3600

Table 1 reports the results from fitting analyses on simulated profiles. Boxplots of the radius and tolerance estimates by LS and MZ algorithms is depicted in Figure 6.

The mean value of LS estimates of tolerance is always greater than MZ estimates. Moreover, it can also be observed that the mean values of tolerance estimates increase with increasing of $C_1$, due to the loss of circular shape by profiles.

**CONCLUSIONS AND FUTURE ISSUES**

As the functional requirements of products become more complicated and the tolerances become more stringent, measurement is one of the fundamental concerns in quality control. Investigations on surface and dimensional errors in turning have received attention in the literature.

The purpose of our research is to study the effects of CMM measurement error on geometric tolerance zone estimation. To such an aim, a model for predicting roundness errors in turning operations is presented in this paper. Measurement uncertainties are simulated through a frequency model. The LS and MZ algorithms have been exploited to estimate substitute features and form tolerances.

One direction of future research is to assess the fitting algorithms for other geometric features: e.g. the line, the plane and the cylinder. Further work regarding this issue is underway.
ACKNOWLEDGEMENT

The work described in the paper has been partially funded by the Ministry of Instruction, University and Research of Italy (MIUR), project PRIN2002 – cod. 2002091159_002 “Measurement strategy optimisation in geometrical inspection by coordinate measuring machine at shop floor level”.

REFERENCES


Moroni, G.; W. Polini; and Rasella M. 2003. “Roundness measurement of turned parts” In 6th AIMI. Associazione Italiana di Tecnologia Meccanica Conference (Gaeta, Italy, Sept. 8-10)

<table>
<thead>
<tr>
<th>$C_t$ (% of nominal radius)</th>
<th>radius mean $10^3$ (µm)</th>
<th>radius std. dev. (µm)</th>
<th>tolerance mean (µm)</th>
<th>tolerance std. dev. (µm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 %</td>
<td>LS 2.5401 5.8312e-4</td>
<td>2.544e+2 2.741e+1</td>
<td>MZ 2.5449 1.0065e+1</td>
<td>2.4336e+2 2.8299e+1</td>
</tr>
<tr>
<td>5 %</td>
<td>LS 2.5416 1.7753e-2</td>
<td>2.7705e+2 3.6747e+1</td>
<td>MZ 2.5457 1.2971e+1</td>
<td>2.5648e+2 3.5759e+1</td>
</tr>
<tr>
<td>10 %</td>
<td>LS 2.5463 5.3408e-2</td>
<td>2.7150e+2 6.0153e+1</td>
<td>MZ 2.5486 2.1670e+1</td>
<td>2.5551e+2 5.6313e+1</td>
</tr>
<tr>
<td>15 %</td>
<td>LS 2.5541 1.2280e-1</td>
<td>4.1409e+2 5.5430e+1</td>
<td>MZ 2.5557 2.4171e+1</td>
<td>3.9505e+2 5.0855e+1</td>
</tr>
</tbody>
</table>

Figure 6 – radius and tolerance estimates with the increasing of $C_t$ amplitude (simulation data)
DESIGN AND OPERATION OF AUTOMATED MATERIAL HANDLING SYSTEMS FOR IC WAFER SEMICONDUCTOR MANUFACTURING

Jairo R. Montoya-Torres  
Centre Génie Industriel et Informatique  
Ecole des Mines de Saint-Etienne  
158 cours Fauriel  
F-42023 Saint-Etienne cedex, France

Leon Vermarien  
STMicroelectronics  
300mm Manufacturing Programs  
Z.I. de Roussel BP 2  
F-13106 Roussel cedex, France

Jean-Pierre Campagne  
Laboratoire PRISMa  
INSA de Lyon  
20 avenue Albert Einstein  
F-69621 Villeurbanne cedex, France

Hélène Marian  
Centre Génie Industriel et Informatique  
Ecole des Mines de Saint-Etienne  
158 cours Fauriel  
F-42023 Saint-Etienne cedex, France

KEYWORDS  
Semiconductor fabrication, automated material handling, simulation-based approaches, survey.

ABSTRACT

The wafer semiconductor fabrication process is one of the most complex processes ever. With the advent of the new 300nm wafer generation, semiconductor factories (fabs) require a higher level of process automation. This increase in the natural complexity of the manufacturing process, as well as ergonomic, productivity and economic reasons, have allowed IC makers to pay much more attention to the material-handling topic. In recent years, researchers have been very interested in studying new methodologies for the design and operation of AMHS in the semiconductor industry, employing both simulation-based models and hybrid OR-simulation approaches. This paper surveys those approaches and identifies opportunities for future research. This analysis has general applicability as a reference tutorial for practitioners and researchers working with the semiconductor industry.

INTRODUCTION

Semiconductor manufacturing is one of the most complex production processes ever. Its complexity is mainly due to two factors. On the one hand, semiconductor manufacturing is a hi-tech environment characterized by huge fixed costs for the required machinery, equipment and tooling (Uzsoy et al. 1992,1994). In addition, machinery rapidly wears out and the technological development is high, so to frequently request new planning criteria. Hence, in order to achieve a satisfactory level of return on shareholders' equity, wafer fabrication facilities (hereafter referred as fab) are usually run on the basis of three shifts per day, seven days per week, 52 weeks per year. Under these conditions, often the utilization rate reaches 95% with reference to the up-time of the bottleneck work centres (Cigolini et al. 1999).

On the other hand, with reference to the traditional job shop context, the majority of the additional complexity lies in the ‘re-entrant’ nature of jobs routing (Kumar 1993). In fact, a typical semiconductor fab contains hundreds of various machines and tools such as masks used for lithography. Most machines are designed to carry out several similar processing steps during the whole processing sequence and for multiple products. Machines of the same type are usually grouped into work centres for several reasons: reduction of set-up time, redundancy in case of breakdowns, efficiency utilization of operators, and having backup when maintenance work is done (Cardarelli and Pelagagge 1995). Since the number of operations that have to be carried out exceeds the number of available machines, several of these operations are done at the same work centres. This means that a lot visits a work centre more than once (in fact for modern technologies, up to 40 times). This feature leads to a system configuration where, at the same time, there are queuing at a machine centre several identical parts, with reference to the job type, but at different levels of completeness with reference to the position on the job’s routing. In re-entrant production systems, most of the basic concepts of operations management have to be otherwise considered.

As a result, considerable amount of research has been done in the literature to analyse some key problems of the production system, such as scheduling of production activities and factory layout. In addition, with the advent of the 300mm wafer generation, new wafer fabs will have a higher level of automation. Hence, an automated material handling system (AMHS) is now required, not only for ergonomic and productivity reasons (Paprotyn et al. 2000), but also in order to maintain the economic viability of 300mm fabs (Murray et al. 2000). So, wafer manufacturers pay much more attention to the material handling topic and, in addition to the classic production scheduling and dispatching problems, they are now confronted to two other problems: (i) the selection and design of an AMHS, and (ii) its management in accordance with production goals (i.e. short manufacturing lead time, high throughput, machine utilization). Since the handling of material (i.e. transport of wafers between manufacturing operations) is a non-value added activity (but without it no value can be added), it is important to note that having the AMHS being the bottleneck of the manufacturing system is unacceptable.
In such a context, the purpose of this paper is to present the state-of-the-art research on the operation of automated material handling systems including its design and vehicle control strategies. In the following sections we will review significant research efforts and we will examine those approaches in order to identify opportunities for future research.

**AMHS LAYOUT DESIGN**

The design of efficient material handling systems in wafer fabs mainly concerns two interrelated problems: (i) the design of the material flow network that provides the resource interconnections, and (ii) determining the size of the vehicle fleet. With the increase in factory automation, the shop design problem has been addressed with various degrees of success. Transport operations in wafer fabs have usually been classified into inter-bay and intra-bay transfers (Aoki 1990). Most of the work presented in the literature is related to design of inter-bay layout configurations, but some works also analyse intra-bay material handling. Nazzal and Bodner (2003) propose a general methodology to tackle the problem of designing AMHS in 300mm wafer facilities. The proposed framework divides the design process into architectural and elaborative levels. Prior to the design stage, fab data are preprocessed using simulation of manufacturing operations. Both output and fab requirements data are then profiled to aid in decision-making at the architectural level. Once those decisions are made, lower-level design decisions are made and analysed using a simulation model that incorporates the AMHS.

**Inter-bay Material Handling Systems**

The objective of automated inter-bay material handling is to transport material from one production bay to another without the intervention of human operators. Typically, the system configuration consists on automated storage and retrieval system (AS/RS) or stocker located in each production bay, which is sized for all material inventory in that bay (Pillai 1990). In the literature, various works have investigated different configuration of AMHS.

Peters and Yang (1997) investigated spine and perimeter layout configurations of AMHS using integrated approach. An AMHS with spine configuration typically has one directed flow loop and crossover turntables for changing travel directions, while a perimeter layout is typically designed with two physically separate loops to form unconstrained two-way travel around the perimeter of the facility (Carpenter et al. 1993). In Peters and Yang’s heuristic procedure, for a given fab layout, the AMHS design problem is formulated as a network flow model. The authors determine an initial solution by considering some feasible initial layout sequence. The steepest-descent-pair wise-interchange (SDPI) heuristic is then used to improve the quality of the solution by evaluating cost reductions resulting from exchanging two departments in the flow sequence. A drawback is that the solution given by the SDPI heuristic may be sensitive to the initial layout. To solve this, Yang et al. (1999) proposed a hybrid taboo search-simulated annealing algorithm that is less sensitive to the initial flow sequence and more efficient with respect to computational time. Additionally, for the spine layout configuration, an integrated approach based on a modified quadratic set-covering problem (QSCP) may be used (Yang and Peters 1997). This procedure exploits the special configuration of the semiconductor fabs to determine the optimal AMHS layout design. The model considers that the cost is directly proportional to the distance travelled. Hence, the objective is to develop a good layout and material handling system design (i.e. location of stockers) that minimizes the material movement cost.

Other inter-bay configurations such as circular layout or bi-directional spine layout have also been studied (Ting and Tanchoo 2000, 2001). These authors use a combination of Operational Research (i.e. dynamic programming, graph theory) and Discrete-event Simulation techniques to solve the problems. Concerning the technology, various authors have been interested in comparing overhead (i.e. zero footprint system) and continuous flow (i.e. conveyor system) material handling (Paprotyn et al. 2000; Schulz et al. 2000). These simulations, however, consider a relatively small semiconductor facility and the impact of AMHS failure is not considered. This point is essential, for instance, in a conveyor system, simply because the AMHS has to be designed in such a way that a segment that is down should not bring down the entire AMHS, and thus greatly increase delivery times.

**Intra-bay Material Handling Systems**

Intra-bay material handling relates to material movement between storage systems (where lots wait) and process equipment (Pillai 1990). This assumes that there are pick-up and drop-off points at each automated process equipment where the AMHS can deposit loads to be processed and pick-up completed loads. Despite its importance in industrial situations, little work has been done to date in the literature, and that work considers particular intra-bay configuration for specific wafer fabs (Kempf 1997; Haagh et al. 1998; Mackulak et al. 1998; Campbell et al 1999; Mackulak and Savory 2001).

To improve the performance of AMHS in a 300mm wafer factory, Lin et al. (2003a,b) proposed the connecting transport concept, using a different type of vehicle between bays than within bays and a single system of interconnected lines. In this connecting transport system, the time that is spent waiting for an empty vehicle is effectively eliminated, and the work-in-process inventory level can be reduced. These authors also investigated the relative performance of the connecting transport AMHS, by comparing the use of different types of vehicles for the same configuration of AMHS layout. The study concluded that, although the combination of vehicle types has a significant effect on average travel time, throughput rate and vehicle utilization metrics, no one of the tested methods outperformed the others in all operational scenarios. The paper also proposes a methodology to predict the system performance based on different values of the number of vehicle and the material flow rate.
Table 1. Summary of Current AMHS Control Strategies

<table>
<thead>
<tr>
<th>Reference</th>
<th>AMHS configuration</th>
<th>Strategy</th>
<th>Objective function(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rooney et al. 2001</td>
<td>Inter-bay</td>
<td>FIFO</td>
<td>Moves per time-unit</td>
</tr>
<tr>
<td>Lin et al. 2001</td>
<td>Inter-bay</td>
<td>FEFS + nearest vehicle</td>
<td>Transport time, Waiting time, Throughput, Carrier utilization</td>
</tr>
<tr>
<td>Jang et al. 2001</td>
<td>Inter-bay</td>
<td>Look-ahead</td>
<td>Flow time, Carrier utilization</td>
</tr>
<tr>
<td>DeJong and Wu 2002</td>
<td>Inter-bay + Intra-bay</td>
<td>FIFO</td>
<td>Delivery time</td>
</tr>
<tr>
<td>Jimenez et al. 2002</td>
<td>Inter-bay + Inter-floor</td>
<td>FIFO</td>
<td>Delivery time</td>
</tr>
<tr>
<td>Montoya Torres et al. 2003</td>
<td>n.s.</td>
<td>Linear programming</td>
<td>Transport time</td>
</tr>
</tbody>
</table>

Capacity Planning

Within the context of AMHS design, the capacity planning refers to the problem of determining the number of vehicles necessary to transfer material between the resources on the shop floor. Little work has been done in the literature concerning the design of formal and analytical methodologies to determine the number of required vehicles in AMHS for semiconductor fabs. Rooney et al. (2001) explicitly calculated the maximum transport capacity of the system by using a static model of a multi-product multi-process wafer fab. Once the number of required vehicles was determined, system dynamics (i.e. waiting times, non-deterministic loading of machine outputs) were incorporated in the model using a dynamic representation of the fab. A sensitivity analysis of the number of transporters permitted to determine the influence of this factor on system’s behaviour. The work of Sturm et al. (2003) investigated a methodology for robust AMHS configuration planning based on congestion and system performances metrics. The rest of the published works uses the fleet size as a factor on the simulation model’s experimental design to determine a pseudo-optimal fleet size and to principally analyse the sensibility of performance metrics.

MANAGEMENT OF THE AMHS

For a predetermined AMHS network path and a given set of vehicles, the AMHS control management problem consists on the assignment of the loaded and unloaded moves to individual transporters. This problem shares many common attributes with the vehicle routing problem in metropolitan networks (Golden and Assad 1988), for which powerful solution approaches have been developed. However, the distinct characteristics of the vehicle routing and scheduling problem in production environments call for specialisation of solution procedures. Broadly speaking, the management of automated material handling systems basically consists of two different, but interrelated, problems: scheduling and routing of vehicles. The scheduling of vehicles consists of assigning a fleet of vehicles to a set of transport requests by jobs (i.e. lots of wafers). The vehicle scheduling decision has to be taken so as to optimise an objective function (i.e. the time a job waits for transport). The routing decision concerns three aspects (Qiu et al. 2002). First, it should be determined if there exists a route conducting the vehicle from its current location to the pick-up point (i.e. machine, stocker) and then from there to the final destination. Secondly, the route should be feasible (i.e. conflict- and congestion-free). Finally, the route should be optimal, or at least pseudo-optimal according to an objective. Depending on the type of service request, the transport of a lots in wafer fabs may be done in one of the following manners: (i) from a machine to the bay’s stocker, (ii) from a bay’s stocker to a machine, (iii) from a machine directly to another machine, or (iv) from a stocker to another stocker. In the general literature, considerable amount of research has been done in the recent years. A state-of-the-art survey can be found in (Qui et al. 2002). In the semiconductor industry, however, little work has been published (see table 1), although the importance of an intelligently managed AMHS.

Basically because of the production process complexity and the rapid market evolution, wafer semiconductor manufacturers have traditionally employed common-sense heuristic algorithms for the planning and control of production activities. In automated material handling system, the most used rule has been FIFO (first-in-first-out) or some of its variants. FIFO rule assigns vehicles to service requests in the same order that lots call for a vehicle. This rule is used in the works of Rooney et al. (2001), DeJong and Wu (2002) and Jimenez et al. (2002) for inter-bay lot transportation. A disadvantage of using FIFO is that priorities (i.e. urgency) in the processing of lots on the machine centres are not taken into account by the strategy.

If we consider the AMHS as an isolated system, its behaviour critically depends on the vehicle dispatching rule employed. Lin et al. (2001) tested various dispatching rules and showed that they have a significant impact on average transport time, waiting time, throughput and carrier utilization. The paper shows that the combination of the shortest distance with nearest vehicle and the first-encountered-first-served (FEFS) rule outperformed the other rules tested. Jang et al. (2001) tested a look-ahead strategy for inter-bay transport in an AMHS with AGVs. Simulation results based on the analysis of average flow time, vehicle utilisation and metrics about the storage system suggest that this strategy may be effective in systems when the increase in the fleet size does not improve factory performance metrics.

The presented works use simulation methodologies in order to identify the conditions in which tested rules perform well. Montoya Torres et al. (2003) proposed an analytical study by formalizing the problem of scheduling wafer transporters using a linear programming approach. In their
mathematical formulation, the vehicle-scheduling problem is solved for a given set of machine schedules. Although the mathematical model is “simple”, it does not solve real-life situations with acceptable computational time.

**FINAL DISCUSSION AND PERSPECTIVES FOR FUTURE RESEARCH**

This paper presented a survey of solution approaches on the design and operation of AMHS in the semiconductor industry. In the recent years, the advent of the 300mm wafer generation has pushed wafer manufacturers to pay more attention to the material-handling topic. However, research efforts done to date in the literature only analysed the AMHS layout problem as independent from the facility layout one. Hence, in terms of practical issues, a first opportunity for future research arises from the integration of both the flow path design problem with the shop layout, together with the vehicle scheduling and routing problem. In the particular case of semiconductor fabrication, having the AMHS be the bottleneck of the factory’s production is unacceptable. If high throughput and low manufacturing lead time are desired, appropriate analysis techniques and tools are needed to carefully explore and plan the operation of an AMHS. Additionally, both re-design and enlargement of existing wafer fabs are interesting from both scientific and practical points of view, occurring more frequently than clean-state design.

For the problem of AMHS control, little has been described in the literature for semiconductor industry applications. In contrast, a large amount of research has been done in the industrial engineering and operations research literature for flexible manufacturing systems (FMS). In effect, works on routing and scheduling for automated guided vehicles (AGV) in manufacturing facilities may be applied to any type of horizontal-based material handling. However, these approaches are limited in scope in the sense that many of the existing solutions consider the problems of routing and scheduling vehicles in a production system as one unique problem. Because of their different objectives, these are two different, but interconnected, problems. Furthermore, existing approaches for FMS are computationally restrictive in real-time environments when addressing real-life problems (i.e. when the size of the system increases either in number of vehicles or in the size of the network path) (Qiu et al. 2002). More complex manufacturing systems such as wafer fabs will require suitable algorithms of scheduling and routing wafer carriers, which guarantees that congestion and deadlock problems will not occur. Hence, future research should attempt to bridge these gaps in the existing literature and must provide appropriate performance measures to evaluate the efficiency of the AMHS to contribute to the overall factory productivity.

**ACKNOWLEDGMENTS**

The work presented in this paper was partially supported by the French Association of Technological Research (ANRT) under research Grant number CIFRE-621/2002.

**REFERENCES**


BIOGRAPHY

JAIRO R. MONTOYA-TORRES is a PhD candidate at École Nationale Supérieure des Mines de Saint-Etienne, France. His doctoral research concerns the optimisation of automated material handling systems for 300mm wafer fabrication facilities. He holds the degrees of Industrial Engineer (highest honours) from Universidad del Norte, Barranquilla, Colombia, and Master of Science from Institut National Polytechnique de Grenoble, France. His research interests include modelling and simulation of manufacturing and logistic systems, on-line scheduling and combinatorial optimisation applied to real-life problems. He is member of the European Operations Management Association (EurOMA). He can be contacted at his e-mail address: montoya@emse.fr
A Petri Net Application: Modelling a High Complexity IC Design Process

Raquel Fernández-Ramos, Jorge Romero-Sánchez, Francisco Ríos-Gómez,
José Martín-Canales, Javier Marin-Martín, Antonio I Molina-Conde
nemo@ctima.uma.es, caronte@ctima.uma.es, frios@ctima.uma.es,
sefran@ctima.uma.es, marin@ctima.uma.es, aimc@dte.uma.es

KEYWORDS
Design Methodology, Petri Nets, Independent of Technology

ABSTRACT

We present a design methodology formal description for high complexity digital ICs. The independent of technology stages as well as dependent of technology ones of the method are well defined and structured. For this purpose we have used Petri Nets with conditioned transitions. The design methodology links in a formal way the actual trends in logic synthesis and physical design, with a new stage in the chain of IC design: the high-level behaviour and discrete-time modelling of the system.

INTRODUCTION

Many efforts have been made in integrated circuit (IC) design methodologies, all of them focus on the automation of the whole design process. Designers always have been constrained with the available tools and the dependence with the technological process, making difficult the task of designing, and practically impossible the translation to other technology and reusing the system when the complexity of the designed IC grew up. because of appearing of high-level design tools, the concept of independent technology design became more and more realistic. Such tools decouple the task of developing the IC from the concept to the layout, speeding up the whole design process, reducing the time to market, also letting the designers to optimize in a shorter time the system, and giving the possibility of changing the technology in a relatively easy way, complex systems included.

The tools we refer are high-level simulators using standard programming languages, HDL languages and simulators, high-level logic synthesis connected with the usual low-level environments such as schematic editors, place & route tools, and physical simulators. The management and scheduling of this tools and how all resources can be linked yields a design methodology. Many design methodologies have appeared during this years, always adapted to the available tools and technology and tending to converge fairly in a few years. Our interest in the IC designing processes, had led us to propose solutions to this important topic, focusing mainly on independent of technology ICs, and the framework surrounding them.

In this work, we present a formal description of a design methodology for high complexity digital ICs based on Petri Nets (PN). Under this framework the whole description of the IC is modelled, from the conceptual level to the physical layout. Several design tools are involved in the process of design, they are commercial except a new design tool for discrete-event simulation based on PN and focused on the analysis and design of independent technology Digital systems, in general: SimulPet, which is used to model the behaviour and the discrete-time ordering of operations of the IC. This strategy has been successfully applied to the design of a high complexity IC (System on Chip) CodeCrypt: a Low-Rate speech CODEC with Encryption capability, actually embedded in our Internet Protocol-Phone (IP-Phone) System running under H.323 protocols.

The article is organized in five Sections. Section I, introduces the concept of Technology Independent Integrated Circuit, and we show the advantages obtained when we design with independence of technology, especially in high complexity digital IC. In section II, the model used to describe the methodology is introduced, we deal with PN. The terminology surrounding such tool is briefly explained, and how a design methodology can be modelled using PN. Design methodology is introduced and explained in Section III, based on its PN model, each stage or design process is explained in more detail, with emphasis in the independent of technology stages, concisely the new one: Behaviour and discrete-time modelling, in which we have based to develop the methodology, and connecting then with a more standard and well established methods: logic synthesis and physical design. A real example illustrating this methodology is showed in Section IV, the digital IC CodeCrypt, a high complexity Low-Rate speech CODEC-Encryption system which has been successfully developed under this framework. The final conclusions we obtain are showed in Section V. To follow the explanation, Figure 1 shows the flowchart of the methodology we propose, it will serve as reference during the rest of the paper.

I. THE CONCEPT OF THE TECHNOLOGY INDEPENDENT INTEGRATED CIRCUIT: TFC

We can define a TFC as a system developed with the aid of specific high-level tools and languages appeared recently (around ten years ago). The result is a design whose behaviour and functionality are only described in formal way, i.e. by using a hardware description language (VHDL,
Verilog, etc) and with independence of the technology we use. The concept of physical IC is not present at this level, although the digital architectures and netlist are already defined.

The independence of technology (IT) leaves the designer to play with a series of advantages very difficult to apply when a dependent of technology (DT) design is developed, especially when the complexity of the system grows up (Romero et al. 1999) (Hillen 1992) (Wang et al.). Such advantages are:

1. **Capabilities for exploring the field of solutions of the IT design in a shorter time.** With the aid of special simulators at this level of abstraction, the simulations are very fast, and it is easy to design in parametric way (for example, the word length of the arithmetic as a parameter to explore the quantification error effect on the performance of our system) even if the design is very complex (like in DSP applications) it is not difficult to make changes in the architectures (the parameter in this case is a set of architectures) of the design without changing it behaviour when it is necessary (for instance, we can select between a sequential multiplier if we need to minimize area or to invoke a combinatorial multiplier when we need high speed multiplications).

2. **Technology translation is fairly trivial.** All designers know how difficult is to translate a design from one technology to other (many times this task is impossible) for adapting the system to new environmental specifications, to reduce power, area, etc; or simply because the actual technology has disappeared. In a IT design we have an unmapped system ready to be implemented in the technology we want depending on our actual needs.

3. **Time to market is reduced** because no redesign is needed. This is a consequence of ii) and, as we will see, if an accurate modelling is achieved in the IT stages of the methodology, aided with fast simulators, the mapping process of the design into a technology and the physical design becomes faster.

4. **Reusing the design is made easy.** i.e. we can use our previously designed system as a forming subsystem of another one, obtaining the new functionality in shorter time than in DT designs (this is consequence of iii).

5. **IT design tools are available at this time.** Although such tools are in continuous development there are several environments for IT systems design. Currently, we depend on how this designs are constrained by these tools during the developing stages of the IT design although, as we will see, it is a drawback.

### II. MODELLING THE METHODOLOGY.

We have followed a very general, intuitive, and widely used modelling technique: a high-level Petri Net with boolean conditions in the transitions. In the next paragraph we explain briefly the terminology used with PN, and how it can be executed. (A formal description of PN theory can be studied in (Peña and Cortadella 1996), (Zhou 1992),(Zhou and Di Cesare 1993).

A PN with boolean conditions in its transitions is a graph with four classes of elements: *Places*, the circles; *arcs*, the arrows; *transitions*, the rectangles; and *tokens*, the circles inside the places. The set of all tokens in the PN is called *marking* of the PN. We connect a transition with some places with any number of arcs arriving (input places) or leaving (output places) the transition from/to those places. We can add a boolean condition to fire potentially a transition if the condition is ‘true’, or block the transition if the condition is ‘false’. We say that a transition can be fired if the boolean condition is ‘true’ and each input place has a number of tokens equal or greater than the number of arcs connected from the input place to the transition. After a transition is fired the marking of the PN changes: we add/subtract tokens to/from the output/input places in a number equal to the number of arcs joining the input/output places with the fired transition.

PNs can be used in all those systems in which changes or events happen in a discrete, or countable way, for this reason they are called *discrete-event systems*: v.g. population conducts, cryptographic systems, multiprocessor systems, asynchronous circuits, manufacturing systems, etc, see (Movei and Lippman 1997),(Peña and Cortadella 1996),(Lindemann 1998). A process in the system can be associated to each place. The token represents the entity that produces the change in the system (an event), although it can also represent a resource that is needed to execute a process. Time can be associated to a place (or transition depending on authors) representing the duration of the process, such PN is called PN with time.

A design methodology can be considered as such discrete-event system, with places representing the stages, or processes of design with the associated times, and tokens modeling either the design or the resources used to develop and implement the design.

### III THE METHODOLOGY.

We observe, in fig.1, there are two flows, the main one representing the normal path we need to develop the system, and the feedback path needed to achieve the necessary degree of refinement, optimization, or accuracy in the modelling of the system until the final layout is reached; both of them are aided by CAD tools: the resources. The main flow can be divided in several stages showing several levels of abstraction of the model description. The first stage corresponds to the highest level, the conceptual one, and the last stage, the lowest level, the physical one. We use in each stage a set of proper simulation tools not always available, because either they do not exist, or they are difficult to achieve, clearly they have been strong constraints in the IC design during last years.

Such division of the problem in stages, or processes, let us an affordable method for its treatment thus giving us several advantages:

1. **Design is afforded gradually**: from the concept, the start stage, to the layout, the final one, the IC is modelled, simulated and co-simulated, stage to stage in such way that the functionality, the main specification, is preserved. Concluding, there are not wide gaps between design translations from the previous level of description to the following one.
thus warranting, with a high degree of probability, the success of the design.

2. Decoupling of simulation process during the development of the IC in different stages has a very important property: the higher the level of the stage we are working, the lower the time of simulation used. For this reason if we achieve a high of accuracy in the description during the early stages, say the IT stages (previously the logic synthesis), the iterations we need to perform in later stages, i.e. DT ones, will be, in general, lower. It is easier to modify architectures in C or VHDL code and simulate the new one, than to do this at gate-level schematic.

We remark that in IT section of PN we propose a new stage, not found in other methodologies (see (Romero et al. 1999)): the behaviour and discrete-time modelling one. The other stages, i.e. from logic synthesis to final layout, are at this moment well established see (Fernández and Romero 2004), (Synopsys 2004).

**System concept: High-level Specifications.**

This is the first IT stage corresponding to the highest level of abstraction (the first main flow place in our PN model). During this early stage the functionality of the system is specified, and, perhaps, the first time and environment specifications, for instance, if we are going to develop a noise generator we need to know basic time specifications, say sampling period (or frequency), band-width for digital filtering purposes; also we fix the functional specifications as the mathematical model we use, for example, pseudochaotic model, linear-feedback shifting, etc; and the environment specifications, for example we need our design works in a cryptographic system or simply as a general noise generator; etc. For all these kind of specifications we call high-level specifications.

**Mathematical Functional Modelling.**

This is the second IT stage, modelled with the second main flow place in the PN. Here we select the mathematic operations to achieve the functionality fixed in the previous stage, and perform the first simulations, totally independent of time, without thinking about neither digital architectures nor timing structures. Usually, like in DSP applications, the data to process is given in batches, and no real-time process is involved. Floating-point arithmetic is used to get a reference for later comparisons and performance measures purposes during the quantifying of the algorithms. The resources used are simulation environments are based on high-level programming languages. We have used Matlab 6.0 for this purpose.

**Behaviour and Discrete-time Modelling.**

This is a very important new IT stage we propose, intermediate between high-level and medium-level of abstraction, that is not included in other design methodologies in which there is simply a gap. Here, we take the functionality and we translate it in basic digital operations describing the behaviour of the system. It consists in building the equivalent quantified digital architectures, and ordering the events that occur in the system in discrete-time say, without considering the absolute time or the absolute duration of a process, but the order of occurrence of the event. This ordering does not correspond only to the sequence of operations but also the more general case: concurrence or parallelism of operations. In logic terminology this concept corresponds to asynchronous systems being synchronous systems a particular case. The absolute time will be backannotated during feedback of timings from posterior stages in successive iterations of the method, if it is necessary. An appropriate modelling tool to use at this stage is PN with time, here we have to add to the advantages obtained with modelling with PN another ones, these are the fast simulations speeds we can achieve, the high degree of accuracy in the description of the model (which depends on the needs, knowledge of the problem and skill of the designer) and the solid mathematical background surrounding them. Basically in our scheme, any digital system could be modelled, in any degree of depth, in a control unit, the own PN, and a data processing unit, the process associated to each place.

The resources used are more specialized than those employed during the mathematical modelling. In our case we need a PN with time simulator basically to model and simulate the high-level design. There are several PN simulators and tools, but we have developed and used our own simulator SimulPet (Fernández and Romero 2004). This simulator that can manage any size stochastic PN with variable time (capability useful for variable rate systems) and logic conditions in the transitions. SimulPet is dedicated to the analysis and design of digital systems, in general. SimulPet is dedicated to the analysis and design of digital systems, in general. SimulPet let the user to model non-stationary Markov-chains, an interesting capability for studying, for example, noisy digital channels in the general case. The tool also has graphical capabilities: animated-PN, high-level timings, very useful to study concurrence in either parallel or pipeline systems, performance index calculations: throughput, speed-up, BER, and any use-programmed index, etc.

Once desired model is obtained with the same functionality, and no feedback to the previous stage is needed, then we can fire the corresponding transition by asserting the OK condition to follow the main path.

**Bittrue Translation.**

We translate during this stage, the PN structure, i.e. control units: basically a set of flags, counters, and some combinatorial logic; and data processing units: adders, multipliers, etc; to a bittrue representation with a standard netlist format. For example, a variable modelled in the behaviour stage could represent in bittrue translation a register, a wire (connection) or a multiplexer; an if-then-else could represent a comparator and a multiplexer; or a loop could be translated into a counter, a comparator, and so on.

This process can be realized, actually, automatically for synchronous systems, but the general case, say asynchronous logic, is not available at the moment, because there are not tools to support the general case, here our dependence in tools is clear. Thus our very general model need to be particularized to a synchronous system at this stage. Once we have a bittrue representation we can add another specification to our system: the clock period. From this time specifi-
cation we can derive all time specifications (except the DT time specifications: the gate delays and sequential logic timings, i.e. set up and hold times basically) multiplying this quantum unit by the discrete-time duration of each process. These times are backannotated (by asserting the NOK condition in the PN) to the previous stage.

Figure 1: Schematic of the Proposed Methodology, Modelled with a Petri Net.

The resources are restricted to synchronous design as we mentioned: we have used an stream driven simulator, CoCentric from Synopsys (Synopsys 2004), for building the bittrue model and generate automatically the VHDL code and a VHDL simulator VSS from Synopsys too. Here, the functionality is again compared with the previous one to assure the correct translation of the system.

The bittrue representation at this level is the final IT stage, and is the main objective of IT designs: an VHDL (or other hardware description language) representation of the integrated circuit: the TFC.

**Synthesis: Low-level Specifications. Technology Mapping.**

Synthesis is the first DT stage in which we must select the technology from available manufacturers and then fix the low-level specifications: Voltages and current of the logic: TTL, CMOS, ECL signal-levels, real environment of our IC: temperature, current driving, parameters of the clock, capacitive loads, wire loads models, etc.

The criteria followed in the selection of the technology depends on power, area, timing, and cost requirements. These requirements are the goals to be met by the synthesizer, subjected to several DT constraints, for example maximum fan-out and capacitance, and DT design rules, as well. The synthesis procedure with floorplanning management closes the synthesis loop in this methodology. In this procedure we start with the physical models (wire-load and delay) given by the manufacturer we selected, then we perform a relaxed goals synthesis, and a gate-level VHDL simulation to check the functionality is maintained (these can be seen in the methodology graph), if OK, we can do an initial floorplanning (preplacement) to extract a first estimation of the delays, because there is not routing paths yet, we can not extract neither net capacitance nor resistance. These delays are feedback to the synthesis tool (following the feedback path in the PN) and a new synthesis with the backannotated delays can be performed with more exact information. When no more gain is achieved in the synthesis process then stop the procedure i.e. convergence is met. We could feed back more exact information if we realize after floorplanning a true placement a route process to extract capacitance and resistance. This case depends on our goals and requirements. The final design at this stage is a schematic described in a standard form.

The tools used in this procedure and at this stage are a synthesis tool with floorplanning management capabilities and a gate-level VHDL simulator. We have used the Synopsys Design Compiler and VSS Expert (Cadence 2004).

**Floorplanning.**

At this point, we start the first low level stage led by an schematic of our design. We can perform the normal and well established steps during these last stages. Remember that if this stage are coupled with the previous one, we can optimize our IC until match our design goals. Basically, in a preplacement we select the bounds of the cells in the die with fixed geometry and area. We can group our cells either following the hierarchy of the design, or simply performing a flattening of the whole design. The delays, if floorplanning management is performed, extracted from this early placement can be backannotated to the synthesis stage or simply we can follow the main path till the next stage.

There are several tools involved with physical design of IC, some of them have incorporated applications above the floorplanning tending to a higher level design tools, but actually, they aim mainly in physical design. We use Cadence environment (Cadence 2004) for these three last
stages and HSpice for physical simulations of the final IC.

**Placement and Routing.**
These are the lowest level stage in which designer can operate. They consist basically in the placement of the physical cells in the die, followed by the wire routing for connecting these cells. For complex designs these task usually are performed automatically and a final physical simulation can be done (very expensive from a point of view of CPU and memory use), here net capacitance and resistance of the wires routed can be extracted with a high degree of accuracy. They can be feedback to the synthesis level for optimization purposes, as we explained.

**Layout.**
Once this stage is reached a physical layout of the IC is built for manufacturing, and the general procedure is closed. This is represented by the exit place of the PN.

**IV. DESIGN APPLICATION.**

This methodology has been successfully applied in a high complexity IC: CodeCrypt (Spanish patent n. 9800213), a low-rate speech CODEC with Encryption capability. This IC has a fully integrated low-rate, low-power, full-duplex Vocoder, based on a LPC model of voice, with user-programmed transceiving rate (3500, 6500, 9600b.p.s.), more than a thousand million key field (32bits key length), and the possibility of silence-compression selected under user-control. The IC has three modules: a Speech-Analyzer, Speech-Synthesizer and a Serial Interface-Adapter. A total of 17 high complex DSP modules, which includes for example, the Levinson-Durbin algorithm for filter coefficient extraction, lattice IIR filter with programmable coefficients for synthetic voice reconstruction, or an Amplitude-Difference of Magnitude Function (AMDF) for pitch extraction. The rest of the modules (no DSP modules) implemented includes the encryption-decryption system, variable-length frame serial transceiver, watch-dog timer for time-out control, and a basic synchronization-protocol embedded in the Interface-Adapter. An extra capability for Academic and testing purposes is the programmed topology (variable structure of the system) of the whole system, for individual testing and evaluation of each module, which can be accessed individually from a program.

The whole system, i.e. two Codecrypt ICs connected through a basic noisy channel, has been modelled with stochastic PN with time and simulated with SimulPet program see [1], for functionality and robustness studies. The IC has been implemented in ALCATEL Microelectronics CMOS 0.7µm. Finally, CodeCrypt is the kernel of a IP-Phone: TiePhone. Existing two products: software version and hardware version (Tie2C design is clear here). Software version can be found in www.Tiephone.com for evaluation purposes.

**V. CONCLUSIONS**

1. We have presented a formal description design methodology for high complexity digital ICs. The independent of technology stages as well as dependent of technology ones of the method are well defined and structured. Due to the discrete-events characteristic of design methodology it has been modelled under Petri Nets framework, a well established and formal tool for modelling and analysis of discrete-event systems.

2. The design methodology links in a formal way the actual trends in logic synthesis and physical design, with a new stage in the chain of IC design: the high-level behaviour and discrete-time modelling of the system, which cover the gap existing between the conceptual design stage and the bittrue translation stage.

3. Actual trends in digital design focus on independent technology design concepts like time to market, systems on chip, reusability, Intellectual Property, and portable systems are becoming possible with the appearing of new design tools.

4. We have developed a simulation tool based on Petri Nets: SimulPet. The environment is focused on digital systems design in general.

**REFERENCES**


ELECTRONICS SIMULATION SOFTWARE
SECS-II SIMULATION USING SEMULA

Naren Koka\textsuperscript{1} and Chell Roberts\textsuperscript{2}

\textsuperscript{1} Indidge Systems, Tempe, Arizona,
\textsuperscript{2} Department of Industrial Engineering, Arizona State University, chell.roberts@asu.edu

KEYWORDS
Semiconductor Manufacturing, SECS-II, SEMI E5

ABSTRACT

This paper presents a simulation tool for testing compliance of the SECS-II SEMI E5 standard. A background providing context for the application is presented. The context includes a brief introduction to standards for semiconductor automation and the scope of an SECS-II simulation tool. Features of the simulation tool are described including the simulation kernel, simulation validation tools, and smart editing tool. An example simulation scenario is presented that demonstrates many of the simulation tool capabilities. The potential use of an SECS-II simulation system as a foundation for higher level semiconductor simulation is presented.

INTRODUCTION

The increasing complexity of semiconductor manufacturing and a strong need to increase productivity in a global business environment has resulted in the establishment of a number of semiconductor manufacturing standards. Among these standards are those developed through the Semiconductor Equipment and Materials International (SEMI) program. SEMI standards address a broad host of semiconductor manufacturing including equipment automation, communication, and processes. With the implementation of 300 mm wafer technology and the need to integrate equipment and processes from multiple vendors there is also a need to increase analysis of the system components prior to installation (SEMATECH 2000). The standards provide a foundation for developing tools for e-diagnostics, remote fault detection, real-time scheduling, and system level optimization including simulation. Three standards that form the building block from which many future tools could be developed are the SECS-II Semiconductor Equipment Communication Standard 2 (SECS-II), the High-Speed SECS Message Services (HSMS), and the Generic Equipment Model (GEM) standards (SEMI 2004).

The SECS-II standard is maintained and published by the non-profit organization Semiconductor Equipment and Materials International (SEMI). It is designated as SEMI E5 (SEMI 2003a). SECS-II protocol is the standard for the communication between the semiconductor manufacturing equipment and factory automation systems. The protocol is defined in the SEMI E-5 standard. The E5 standard contains data and message formats for communication between equipment and a host. All semiconductor equipment manufacturers are required to support the SECS-II E5 standard on their equipment. The SECS-II protocol is a dedicated host-equipment protocol. Using this protocol, only one host can connect at a time. The SECS-II protocol initially operated on RS 232 lines running at 9600 baud. As the semiconductor automation matured into more advanced technologies, the need for higher speeds, more bandwidth and more flexible protocols arose. A High-Speed SECS Message Services (SEMI 2003b) standard was developed to extend the SECS-II protocol onto Ethernet over the TCP/IP stack. This provided faster speeds required for transferring more data. Equipment manufacturers and semiconductor integrators can test the equipment’s communication features using simulation tools to ensure compliance with the SECS-II standard. The GEM standard, designated as SEMI standard E30, is also published by SEMI. GEM defines a commons set of behaviors and communication capabilities for equipment (SEMI E30). An equipment manufacturer providing a GEM interface allows a factory host to interface with the machine for control and monitoring purposes.

APPLICATION OF SECS-II

Semiconductor chip manufacturers integrate the equipment into their factories using automation software. They use Manufacturing Execution Systems (MES) to control the product flow through the factory. Host applications communicate between the MES and
the equipment by sending instructions and process programs from the MES to the equipment. The host applications also collect status and performance data from the equipment and send it to the MES and other factory systems.

Semiconductor manufacturers and software vendors typically develop the host application software. In order to test and validate their software, a SECS-II messaging system is required. Due to the high cost and complexity of the semiconductor equipment, and the difficulty of implementation in a clean room environment, the use of a SECS-II message simulation tool is desirable. Since all semiconductor equipment manufacturers are required to support SECS-II standard on their equipment, testing equipment communication features is possible using simulation tools to ensure compliance with the SECS-II standard. Likewise, software implementations of requirements for 300 mm testing must be verified for compliance and performance (SEMATech 2000), which can be performed using a simulation tool.

A typical SECS-II simulation tool can be run in two modes – Host mode and Equipment mode. In Host mode, it is used to simulate the host application and is used for testing the equipment. In Equipment mode, it is used to simulate the equipment and is used for testing the factory host applications. Figure 1 depicts an SECS-II simulator running in a host mode. We believe that the characteristics of a good SECS-II simulation tool should include a configurable and robust kernel that can be optimized, message scheduling, configurable data views for validation including compliance with the XML protocol, and robust editing capability.

A message initiated by the equipment or the host application is called the primary message. A message that is created in response to a primary message is called secondary message. Primary messages use an odd number as its function number and secondary messages use an even number. For example, S1F3 is a primary message and S1F4 is a secondary message. Primary messages also contain a flag that indicates if a response is required for that message. When the requestor sends a primary message using an odd function number with the response flag set to true, the receiver sends a secondary message using the next even function number. As an example, when the requestor sends S1F3 with the response flag set to true, the responder sends S1F4.

**The Semula SEC-II Simulation Tool**

Semula is Semiconductor Equipment Communication Standard 2 (SEC-II) message simulation software tool that can be used for the testing and validation of semiconductor manufacturing equipment and factory automation software applications. Semula supports the HSMS standard and the new emerging protocols. Several SEC-II simulation tools are currently available (Asyst, 2004; Mullin, 2004; V1, 2004). Many of these software tools require manual programming and debugging of scripts. An advantage of Semula is that it automates the message creation to reduce the overall programming time and eliminate most of the debugging time.

Semula’s approach is to use internet technologies as a platform for semiconductor factory automation. The approach includes a multi-threaded kernel to handle communication that allows multiple worker threads to process messages simultaneously. Transaction management is performed using a FIFO transaction queue for incoming primary messages. A new primary message creates a Transaction Event Object that is stored in a Transaction Queue. Semula also offers a feature to schedule periodic outgoing messages. This feature permits a simulation to be configured based on a schedule for creating primary messages. Simulated messages are scheduled as transaction events and are stored in the transaction queue along with the incoming messages. The simulation engine processes the internal message requests similar to a discrete event simulation engine.

Semula maintains a pool of software threads, or entities, called workers that service the transaction events. As a worker becomes available, it grabs the earliest
transaction, processes it, and dispatches the response to the requestor if required. The number of workers in the pool is configurable and affects the performance of the simulation engine. This number can be configured as suitable to the specific environment, so that the desired communication speed is achieved by balancing the ratio of workers in the pool to Transaction Events in the FIFO queue. This is depicted in Figure 2. Sensitivity analysis can be performed by examining a range of worker configurations.

Three types of display settings can be used help the validation: filters, message formats, and message counters. Filters restrict the messages that are displayed on the screen to the ones that the user has selected. Using filters, the display can be restricted to the specific set of messages. The format of the message can also be configured such that the user can alter the view the information from a simulation run. Table 1 shows the message formats available for display. Semula uses an internal XML protocol for the SECS-II messages. An XML message is self-descriptive when associated with the corresponding XML schema. It is more readable than the binary format of a SECS-II message. Message counters display only a count of the number of messages that are received. This feature can be used to test the performance of the communication system. It may be also used as a mechanism to identify when to end the simulation run.

Table 1: Supported Message Display Formats

<table>
<thead>
<tr>
<th>Option</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td>XML</td>
<td>Messages are displayed in XML format</td>
</tr>
<tr>
<td>SML</td>
<td>Messages displayed in GW Assoc. SML format</td>
</tr>
<tr>
<td>Binary</td>
<td>Message bytes are displayed</td>
</tr>
<tr>
<td>Header only</td>
<td>Only message header is displayed</td>
</tr>
</tbody>
</table>

A Smart SECS Editor

SECS-II is a binary communication protocol. The binary messages are created by arranging several data elements in a structured format. An abstraction of the binary format is necessary to visualize and manually edit the contents of the messages. The Semula Smart SECS Editor handles the abstraction and helps in the editing of SECS-II messages. The Smart SECS Editor is intelligent and knowledgeable of the standards and displays information from the standards for reference. It also controls the graphical user interface and allows the user to build messages according to the standards and avoids typing mistakes. Figure 4 depicts the Smart Editor.

The SECS-II standard dictates the types of data elements that should be included in a SECS-II message. The standard supports several data types including strings and numbers of various formats and sizes. In many cases, the standard allows several data types for a single data element. The Smart SECS Editor enables the user to choose one of the allowed formats for each data element. The protocol also supports list type data.
elements that contain other data elements. A list can be formed by packing multiple data elements into a single higher level list data element. The Smart Editor allows the user to add data elements to the list and to delete data elements from the list as per the standard.

Figure 4: Smart SECS Editor

AN EXAMPLE SCENARIO

In the following simple example scenario, a host application is required to obtain the values of Chamber Pressure, Temperature and Rotation Speed every 5 seconds and display the values. A requirement of semiconductor equipment is to provide the status of the equipment through status variables. In the SECS-II standards, status variables are identified by SVID (Status Variable ID). SVIDs are the status variables for getting equipment status. Status variables may include any parameter that can be sampled in time such as temperature or quantity of consumables (SEMI E5). Allowed formats for SVID are ASCII, 1, 2 or 4 byte integers and 1, 2 or 4 byte unsigned integers. S1,F3 is the function for getting status variables. The definition of S1,F3 is: S1,F3. Selected Equipment Status Request Description: A request to the equipment to report selected values of its status (SEMI E5).

Equipment vendors publish a list of status variables in their SECS communication manual. From the manual we get the SVIDs for the status variables shown in Table 2.

Table 2: SVID Vendor Table

<table>
<thead>
<tr>
<th>SVID</th>
<th>Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>2132</td>
<td>Chamber Pressure</td>
</tr>
<tr>
<td>2144</td>
<td>Temperature</td>
</tr>
<tr>
<td>2535</td>
<td>Rotation Speed</td>
</tr>
</tbody>
</table>

The steps required to accomplish this task are: Create a new project, Edit S1F3 function to request for the values of the 3 status variables, Schedule S1F3 to be sent every 5 seconds, and Run the simulation and view results. The project creation screens are not shown. The smart editor is used to create the status variables as shown in Figure 5. The figure depicts the selection of one SVID from a list of three created variables. The value is modified to 2132 (Chamber Pressure) on the right hand side of the screen. Each variable is likewise modified. At the bottom of the screen the desired format for viewing the variable is selected. In this case U4, unsigned 4 bit. Once the variables are defined, messages are scheduled. Scheduled messages are simulated requests that are sent to a device. Figure 6 shows the message scheduling screen. In this case S1F3 is scheduled to be sent every 5 seconds.

Figure 5: The SECS Message Editor

Figure 6: Message Scheduler

Once all messages are scheduled, the simulation run is conducted. Here we forgo the run start screen and show the output message screen, Figure 7, that is used to trace the simulation and validate proper communication between the host and the equipment. As noted in the Figure, S1F3 messages and the response S1F4 messages displaying the values of Chamber
Pressure, Temperature and Rotation Speed are shown with XML tags.

Figure 7: Simulation Output Messages

SUMMARY AND FUTURE WORK

Simulation tools provide value in the semiconductor industry, where the equipment and the operations are expensive. When combined with the latest software technologies, simulation tools provide a user friendly environment for the verification and testing of the equipment and software applications. A SECS-II simulation tool that includes a configurable and robust kernel that can be optimized, message scheduling, configurable data views for validation including compliance with the XML protocol, and robust editing capability not only provides a good platform for testing equipment configurations prior to implementation, but also provides a foundation for higher level simulation development.

There are several potential areas of expansion for this SECS-II message simulation system. First, messaging simulation, coupled with an equipment model, could be extended to higher level factory simulation models. Factory simulation models are used for identifying the bottlenecks and for understanding the utilization and throughput. When a factory simulation model is combined with a messaging system, it can be used to simulate the entire factory operations from the MES systems to the station controllers and the individual equipment. A combined simulation model can be used to assess the infrastructure requirements for new factories and for testing new product introductions. This capability would permit analysis of factory performance prior to equipment purchase and implementation.

A messaging simulation can also be extended to an equipment control system simulation. The SECS messaging system can be viewed as a sub-system of the equipment control system. The control system activates the equipment devices and works on the product. These interactions generate events on the equipment that are communicated to the host using SECS communication. The volume of SECS messages depends on the equipment, the control system, the product and the demand from external systems. As semiconductor automation has increased, the load on the messaging system has likewise increased and often impacted the performance of the equipment. As more CPU time is taken away from control tasks, equipment reliability is at risk. In order to identify bottlenecks and to optimize the equipment, a simulation system that includes the control system model and the messaging system are required. The control / messaging simulation system can identify the bottlenecks and help to increase the reliability of the equipment.

REFERENCES


8-BIT MICROPROCESSOR ENCIPHER MODULE  
DESIGNING, SIMULATION AND USING IN RESEARCH

Tomasz Wieczorek  
Department of Electrical Engineering  
Technical University of Koszalin  
75-620 Koszalin, ul. Raclawicka 15-17, Poland  
Email: wietom@lew.tu.koszalin.pl

KEYWORDS  
microprocessor, ciphering

ABSTRACT

In this paper has been presented a process of designing microprocessor’s ciphering module. Hardware ciphering modules are probably the best way to cipher. Presented module has been designed for cipher signals transmission loss research. In the paper has been presented all steps of designing, simulation and realization of the ciphering module. Next has been presented first results of researching.

INTRODUCTION

Protection of confidential information is very important problem in IT. Every day in telecommunication networks a lot of information and data are transmitted. Probably the most of this data should be kept in secret. There is only one way to do it. It is ciphering. This problem is also important in underwater acoustic networks (UAN). It is important, because in UAN unauthorized person can hear a lot of transmitted information. In this case it is very difficult or impossible to use typical radio communication modulations, which can keep information in secret. In UAN many users use the same frequency and it is the most important reason for ciphering. Of course there are a lot of problems in UAN, but it is not a topic of this paper. The main research topic is cipher transmission loss in nonlinear transmission channel (for example in underwater transmission channel). In this case special ciphering devices must be used. Those devices should be light, cheap, should have low energy consumption. And it also should have good possibility to upgrade.

PROJECT CONDITIONS

Presented in this paper ciphering module was made as master thesis. In that time there was not any ideas for using it in UAN, but it is good solution for UAN. The project conditions were:

- Module should be light;
- There should be easy way to upgrade;
- Good availability for electronics components;
- Low energy consumption;
- The lowest costs.

Three different ways to build ciphering module was considered. The first one was to build the device with using digital IC. This possibility was rejected. The reasons for rejecting are:

- High energy consumption;
- To large device size;
- To large costs.

Second proposition was re-programmable IC, for example in VHDL technology. This option was consider much more in details. Some parts of ciphering module was prepared and simulated in VHDL technology. It is good solution for this topic, but in this case this option was also rejected. The most important reason to reject this technology were problems with practical realization of this module. At the time, when ciphering device was designed there were not equipment to realize module at Technical University of Koszalin.

Third possibility is microprocessor’s ciphering module. But there is a question: it is possible to build good ciphering system with simple 8-bit microprocessor? In this case INTEL 8051 microprocessor was chosen. Before this decision short analysis was made. Next is the short review of that analysis.

ANALYSIS

During ciphering few different kinds of mathematical operations are made. The most popular are three of them: substitute, displacement (permutation) and basis mathematical operations (for example XOR operation – it is the most popular operation in ciphering algorithms). All this operations can be realized with simple 8-bit microprocessors, like INTEL 8051. But each of them is made in different way. For example: to realize permutation two blocks of memory are needed and each of these blocks has the same length as cipher data block length. But for substitution two blocks of memory and substitution table is needed. On the other side for XOR operations requirements are much more less. It is basis operation in INTEL 8051 code.

After positive effect of analysis two programs were made. First of them is DES algorithm and second is simple substitute cipher. And next step was simulations of these two programs working.

SIMULATION

This part of designing and testing process was divided into two parts. On the beginning special PC software for INTEL 8051 simulations was used. It is AVSim51 software. In picture 1 screen shot of working simulation is presented.
Using this software it is possible to check program correctly and program working. But it is impossible to check and correct some parts of program, which are responsible for working with other devices like keyboard, LCD display and Input/Output ports. Because ciphering module was designed to working with other device (like PC) using serial port, so it was necessary to check this part of assembler mnemonic too. In AVSim51 software it is only possible to check states in registers, which are responsible for serial port servicing. Apart of this difficulty it is very important part of designing process. By using AVSim51 simulator it is possible to find and correct all errors in assembler code (for example: bad mnemonic, too long jumps etc).

Both, DES and substitute cipher, algorithms were tested in AVSim51 simulator. Some errors, especially too long jumps and some mnemonic errors were found and corrected.

After that part of designing process it is possible to realize ciphering system. But communication procedures were not checked. To check them it was necessary to use hardware microprocessor’s system’s simulator. The DSM-51 simulator was used. The DSM-51 simulator is presented in picture 2.

The DSM-51 simulator is extended microprocessor system, which is used at Technical University of Koszalin for microprocessor’s technology teaching. That system has been used with EPROM memory simulator controlled by PICCOLO software. The DSM-51 simulator has two displays (LCD and LED), keyboard and serial port. Using this simulator it is possible to check communication with other devices procedures. So this part of system designing was made. During this part of simulation some errors in communication procedures has been found. The most important was wrong timer settings. After correct these errors both ciphering algorithm was correct. All tests were made with Norton Commander Terminal Emulator and PC program SZYFRATOR (eng: CODER). The CODER program was wrote for presented ciphering module. To test and control ciphering module.

After pass all simulations ciphering module was realized.

ENCIPHER MODUL DESCRIPTION

Presented encipher module was made in simplest INTEL 8051 application. It uses only internal RAM memory and small (32KB) external ROM memory. The module need 5V DC power supply. The encipher module is presented in picture 3.

ENCIPHER MODULE TESTS

After assembly the encipher module was tested. The tests were divided in three parts. Part one was typical electrical tests like voltages and currents. Second part were tests of compatibility with other solutions. And the last one was testing for errors during work.

All tests were passed.

MAIN RESULTS OF RESEARCH

Presented in this paper ciphering module has been used in underwater cipher signals transmission loss research. The first step of main research was preliminary tests. This step should give answer to question: Is any difference during transmission between different cipher algorithms, especially in transmission errors. That research was made with presented encipher module, simulations of other ciphering modules in AVSim-51 and MATLAB. The general scheme of test stand is presented in picture 4.

Most of ciphers was simulated with AVSim51 software and presented encipher module. Only two ciphers was simulated in MATLAB. On the other side transmission loss and data series comparator was simulated in MATLAB. In principal MATLAB is very good tool for simulation.
General results for simple one-sign ciphers are presented in picture 5. One-sign ciphers are ciphers where data block length is 8-bits – one sign (for example: XOR cipher or substitute cipher).

In picture 5 is shown that there is no different between these ciphers. So cipher algorithm sort has no influence for data transmission errors (for one-sign ciphers). It is possible to make tests and research only for one kind of simple ciphers and results will be representative for other kinds of simple ciphers. But probably for other, more complicated ciphers, these results will be different.

**CONCLUSIONS**

Using software and hardware simulators for microprocessor’s systems making designing process faster. Also it is not necessary to build microprocessor’s systems for all needs and all versions. Most of them can be simulated.

Simulation of microprocessor’s system with AVSim51 software is good way to check and correct all errors in assembler code. Very important is information about kind of error, which is generated during simulation. It is usually impossible to find, when microprocessor’s system is made without simulation. Some kinds of errors (for example infinite loops) are impossible to find when microprocessor device is running.

General results for one-sign ciphers shows that it is sufficient to make research only for one of every kind of ciphers. It is mean: one ciphers with 8-bit data block, one for 64-bit data block etc. And the research results should be representative for all cipher signals.

**REFERENCES**

Doroty Elizabeth and Robling Denning, „Kryptografia i ochrona danych” Wydawnictwa Naukowo-Techniczne; Warszawa 1992.93.


Piotr i Paweł Gałka „Basis of Microcontroller 8051 Programming””, MIKOM; Warszawa 2000. (in Polish)

Tomasz Wieczorek and Marcin Wodziński „Signal Transmission with Ciphering Transformation” – Master Thesis Proceeding, don’t published (in Polish)

**BIOGRAPHY**

**TOMASZ WIECZOREK** was born in Połczyn Zdroj, Poland, and went to the Technical University of Koszalin, where he studied electronics and obtained his degree in 2002. Since 2002 he has been postgraduate student at Technical University of Koszalin and he has been worked at the Department of Electrical Engineering. He carries out research on methods of cipher signal transmission in nonlinear transmission channel.
CONCEPT OF A MEMS PROCESS SOFTWARE SPECIALLY DEDICATED TO THE DEEP X-RAY LITHOGRAPHY PROCESS

Pascal Meyer, Joachim Schulz and Christian Solf
Institut für Mikrostrukturtechnik
Forschungszentrum Karlsruhe
Postfach 3640, D-76021 Karlsruhe
E-mail: pascal.meyer@imt.fzk.de

ABSTRACT

Micro-electromechanical systems (MEMS) incorporate miniature electromechanical components fabricated with processing techniques and equipment originally developed for the semiconductor industry. Due to many processes employed for MEMS manufacturing originating in the semiconductor industry, silicon is typically utilized as a MEMS substrate; CAD tools for MEMS dedicated to bulk micromachining and semiconductor industry are commercially available (for example MEMS Pro from MEMSCAP). However, non silicon materials such as plastic and metal substrates are also emerging in microfabrication. The deep x-ray LIGA process, which combines X-ray lithography with electroplating and molding, is a technique used worldwide for the fabrication of high aspect ratio microstructures. Understanding device fabrication is essential in determining if an optimized device can be produced in a cost-effective manner using deep x-ray lithography process. Design and fabrication software tools are inadequate and unavailable. The concept of a general MEMS process physics simulation capability that addresses key process in the x-ray LIGA process is proposed here.

INTRODUCTION

Efficient manufacturing of MEMS using deep x-ray LIGA process (Hruby 2001) can be achieved by reducing the expensive and lengthy product development cycle times. In fact, manufacturing test structures is very expensive in terms of time and materials. When developing systems, the entire process from component specification to drawing preparation can be handled by the designer, but the specific fabrication techniques must be taken into account for the performance of the final system. For example, optical microsystem structures should be taken into account in the optical design stage as the optical performance can be influenced by these structures (Sieber et al. 2003). The final design should meet the requirements of the fabrication process. The LIGA technique (see Fig. 1), a German acronym consisting of the letters LI (Röntgen Lithographie meaning x-ray lithography), G (Galvanik meaning electroplating) and A (Abformung meaning molding) developed at the Karlsruhe Research Center, offers the possibility of manufacturing microstructures with high aspect ratios, high accuracy from a variety of materials (metals, plastics and ceramics). A fabrication example is given in Figure 1. Deep x-ray lithography allows structures of any lateral design with high aspect ratios to be produced, i.e., with heights of up to 1 mm and a theoretical lateral resolution down to 0.2 μm. The walls of these structures are smooth and parallel. Very sophisticated structures of this type can be lithographically produced only by a highly penetrating, intense, and parallel synchrotron x-ray radiation. Our concept of a general MEMS process physics simulation capability that addresses key process in the x-ray LIGA process is represented in Figure 2. Process simulation includes: mask process, resist technology, irradiation and development steps, electroplating, rework, price, manufacturability, work plans. Design simulation includes: physical function of the component, process design rules. Design and fabrication software tools for LIGA process being inadequate and unavailable (Hahn et al. 2000), the central idea of this project is the creation of a process flow representation, which is a description that can be used for the tasks mentioned before: process design and simulation.

Figure 1: A fabrication example to obtain single metal parts
THE CONCEPT

The LIGA process is composed of different steps each of which has its own requirements. How should my CAD design look, what kind of materials should the manufacturer use, what kind of mask process should we follow to fulfill customer specifications (roughness, feature, cost of the end product), how long will it take to obtain the product, what kind of resist should I use, which synchrotron beamline could I use, etc...? A program named XLiGAProSim which could answer all these questions is under development. It will be composed of two modules:

* design simulation,
* process simulation

which will compliment each other. The goal of this program is to offer the user a graphical user interface (GUI) specially dedicated to deep x-ray LIGA. A simple interface has been built for the process simulation (Figure 3). The user will be asked about his desired final product regarding its characteristic: cost, thickness, tolerance, material. The program will establish the best possible process. This will lead to design rules which must be applied for the design of the layout.

Figure 3: Prototype of the user interface of XLiGAProSim with the DoseSim module of the irradiation and development process step

THE DEEP X-RAY LIGA PROCESS

In the first step, the absorber structure of a x-ray mask is copied into resist layers by using synchrotron radiation. Usually, two types of resist material are used: poly(methylmethacrylate) (PMMA) and SU-8. In the case of the former, under exposure, the molecular weight decreases and it becomes soluble in a developer. In the latter case, there is a polymerisation and the unexposed part is soluble. The spaces generated by the removal of the irradiated plastic material can be filled with metal by electroplating processes. In this way, a negatively patterned secondary metal structure, such as nickel, copper and gold, or alloys, such as nickel-cobalt and nickel-iron, is generated. This technique is used to produce:

* microstructures for direct use,
* molding tools.

DESIGN SIMULATION

The simulation includes the physical function of the component and the requirements of the fabrication process. Existing simulation software like ANSYS\textsuperscript{TM}, ZEMAX\textsuperscript{TM} will be used to simulate the physical function and to establish and verify the layout, designed according to the design rules which will be given by the process simulation. For design rules checking, software like MEMSCAP\textsuperscript{TM} (MEMSCAP 2002) exists but must be modified for the deep x-ray LIGA process.

PROCESS SIMULATION Design rules

The x-ray design rules are a function of the final product. For example, special care should be taken in the design of auxiliary structures when the desired products are in metal (Solf et al. 2004). The purpose of the auxiliary structures are to minimize swelling and thermal expansion effects during the electroplating process, which may otherwise damage or destroy the LIGA structures. The auxiliary structures also promote equal current distribution during electroplating, which again leads, for example, to homogeneous nickel iron growth. The layout should be made as symmetric as possible, this is useful for internal stress compensation and subsequently reduces
the deformation of the LIGA structures. Also, differences in the process chain (synchrotron beam-line, type of resist ...) lead to dimensional differences between the CAD file and the end product. In order to yield the specified lateral dimensions, the CAD file has to be adjusted. The recent accumulation of manufacturing information is helping to implement these rules. For example the minimum size of one structure which could be obtained is different if this structure is positive or negative, and if the end product will be a mold or a single part in resist or metal. This size is also function of the mask process which will be used.

Mask process

Different processes exist to produce the working mask, which its characterised by its membrane type and the thickness of the gold absorber. The type of working mask and its fabrication process is a function of the desired product. Two processes developed at FZK/IMT are implemented in the program.

Resist Technology

Resist technology consist of applying a resist onto a substrate. The standard positive resist is PMMA. A database of resist technology is implemented, for example, the resist technology of FZK/IMT. The resist layer is fabricated by gluing a prefabricated, commercially available PMMA sheet (GS233 from Röhm GmbH) with polymerization glue onto the substrate. This process leads to adhesion characteristics which are known and implemented in the program.

Irradiation and development

LIGA technology needs a synchrotron beam-line to perform the resist exposition and a scanner to move the sample. The mask and the substrate holder are one of the parameter of the process. Substrates could be: 4-inch wafer (Si, SiO2, Al2O3) which could be covered by a metal (Ti, TiOx, Au) or metal block (84x54x8mm). In the copy step of the LIGA process, the deposited dose calculations in the resist are needed. The basic calculi needed for synchrotron beam-line design are related to the spectra characteristics and to the modelling of the optical elements (mirrors, filters, beam-stop). For user-friendliness, a GUI (Graphical User Interface) working in a MS-Windows environment has been built (Meyer et al. 2003), which consists of one menu and different worksheets (see Fig. 3) which follow a LIGA beam-line design, e.g. the source, the optics (front-end window, mirror, filters), the scanning stage (mask, resist), the development process of the irradiated resist, and the time of irradiation. This program could be started directly with XLiGAProSim. For a given beam-line, the parameters of irradiation and development for a given mask and resist system is known. A database consisting of resist system (PMMA, SU-8) and synchrotron beam-line (ANKA, Bessy, Elsa) is implemented in the XLiGAProSim. For another resist system and synchrotron beam-line, the GUI mentioned before can be used.

Electroplating

Electroplating is the key step for the fabrication of metallic micro components. Established routine processes are often referred to by the plating solution. Typically every user performs the base processes with differences in electrolyte formulation and operation due to specific fabrication environments such as the plating apparatus. The parameters here are the desired thickness, the aspect ratio and material and its tolerances. A database taking into account these specifications is also implemented. If the thickness tolerances are too demanding for the electroplating process, some mechanical work such as polishing may be necessary. If the metal individual parts should be picked and placed, the type of etchant which should be used must be compatible with the type of material used.

Stripping

This section concerns only PMMA because SU-8 stripping methods are not fully defined. The step consist to irradiated the PMMA and to dissolve it with ethyl-acetate. A database consisting of PMMA Thickness, synchrotron beam-line (ANKA, Bessy, Elsa) is implemented in the XLiGAProSim.

CONCLUSION

Process support specifically concerning the deep x-ray lithography is under development. The rules should need as little supplementary data as possible and rely basically in process, material and medium parameters. This software will offer these rules, design rules and the manufacturing infrastructure necessary for rapid product development and increased commercial success.

REFERENCES


Modelling and Simulation of a Speech Coding Communication IC Using SimulPet

Raquel Fernández-Ramos, Jorge Romero-Sánchez, Francisco Ríos-Gómez,
José Martín-Canales, Javier Marín-Martín, Antonio I. Molina-Conde
nemo@ctima.uma.es, caronte@ctima.uma.es, fríos@ctima.uma.es,
sefran@ctima.uma.es, marín@ctima.uma.es, aimc@dte.uma.es

KEYWORDS
Petri Nets, Modelling and Simulation, Speech Coding, Integrated Circuit.

ABSTRACT

We present a formal description and simulation of a speech coding communication integrated circuit (IC). These can be performed by using stochastic Petri Nets (PNs) with time, which let us to model the IC with independence of technology, and with any degree of depth we need, achieving highly accurate and fast simulation results. Relevant properties of the PNs are used to achieve main goals in IC design: sleep-lock can be used, for instance, to control dynamical power consumption through the VOX system embedded in the IC. SimulPet is the tool employed for system modelling and simulation based on PNs.

1. INTRODUCTION

Actual trends in integrated circuits (IC) design are dealing with the complete automatization of design process. With the appearing of modern design tools the concept of independent of technology (IT) design becomes more and more realistic (Hillen 1992), the designers can focus on their efforts exploring solutions with faster simulators and modern modelling techniques, reducing thus the time to market of the design. Such high-level tools, let the designer skill becomes less dependent with them, and the creative process is, in this way, enhanced. Our interest in IC process design, had led us to propose solutions to this important topics, focusing mainly on IT designs, and the framework surrounding them.

In this work, we present a technique based on stochastic Petri Nets with time for modelling and co-simulation of high complexity digital integrated circuits IC. Under this framework, a digital system can be fully described from a point of view of functionality, behaviour and time-ordering of operations. As main consequence of these points are the independence of technology description at this design stage, see (Peña and Cortadella 1996), (Romero et al. 1992).

This paper is organized in four Sections. In Section 2, stochastic PNs with time are introduced. Under this framework the IC will be modelled and simulated. The description of the behavioural and discrete-time ordering of operations of a modern communication IC is developed with some detail in Section 3: we focus on the CodeCrypt IC processes management and scheduling, with emphasis in the coupling section of two CodeCrypt ICs. Some important results achieved with this technique are summarized at the end of this Section. As simulation example, in Section 4, we illustrate a typical scenario occurring during the programming of the CodeCrypt system, and concluding in Section 5, with the major results achieved in this work.

2. BRIEF INTRODUCTION TO PNs

We have followed a very general, intuitive, and widely used modelling technique: an stochastic PN with time and boolean conditions in the transitions. A formal description of PN theory can be found in (Lindemann 1998), (Zhou 1992), (Zhou and DiCesare 1993), (Chiola et al. 1993).

If we want to simulate the PN versus time, we will only need the duration of each place (or transition depending on authors) and the arrival time of each token to that place. In the case of stochastic PN, there are conditioner places and conditioned places coupled with the own net, and a conditional probability matrix in which each element $a_{ij}$ represents the conditional activation probability of the conditioned place $j$ from the conditioner place $i$: the behaviour of the net is the same than high-level PNs but the marking updating of a conditioned place depends also on the conditioner place as follows:

1. If a token arrives to a conditioner place we select randomly and following the conditional probability matrix distribution the conditioned place that will be potentially activated (`hot').
2. When a transition is fired and there is at least one conditioned place, these places are really activated (adding the corresponding tokens) if they were potentially activated previously by a conditioner place.

A digital IC, a relatively simple discrete system, can be modelled and simulated under the framework of discrete-event systems, being PN an ideal and very general tool for these purposes. The designer can model in any degree of accuracy the behaviour of the system and perform fast simulations and solutions exploration before translating the system to a lower-level implementation and simulation which are necessary slower.

There are several PN simulators and tools, but we have
developed and used our own simulator: *SimulPet* (Fernández and Romero 2004) a discrete-event simulator that can manage any size stochastic PN with variable time (capability useful for variable rate systems) and logic conditions in the transitions. It is focused on analysis and design of digital systems, in general. SimulPet lets the user to model non-stationary Markov-chains, an interesting capability for studying, for example, noisy digital channels in the general case. The tool also has graphical capabilities: animated-PN, high-level timings, very useful to study concurrence in either parallel or pipeline systems; performance index calculations: throughput, speed-up, BER, and any use-programmed index, etc.

3. **MODELLING A SPEECH CODING IC**

To illustrate how an IC system can be modelled using PNs, we will base on a real system: CodeCrypt (Spanish patent n. 9800213), which is a high complexity digital IC which implements a Speech CODEC with encryption and variable rate of transmission capabilities. This IC has been developed following a modern design methodology, see (Romero et al. 1999) for more details. In this paper, we only focus on the *behaviour and discrete-time modelling* of the communication system, an IT stage of the proposed methodology. The whole system is formed by two CodeCrypt ICs coupled through a noisy communication channel. Figure 1 shows the CodeCrypt system, with details of the Interface Adapter (IA) of both subsystems, and the communication channel as the main coupling element between them. The Speech Analysers and Synthesizers of the ICs, for simplicity, are grouped in several ‘black boxes’. The functionality embedded in the graph is a basic communication protocol with two basic tasks: synchronization of the two ICs and speech frame transceiving. Some error protection is implemented in the IA: parity checking and re-synchronization, in the case of parity error during synchronisation process. Functionality description of the system can be summarized as follows:

1. ICs are in Power-Down mode (sleeping) after a Reset sequence, *i.e.* initialization of all registers in the IC. During this time, the users of both systems can program them: say Key Memory, Watch-Dog Timer, transmission rate, etc. The initial rate is 3500 b.p.s and cryptosystem is disabled.

2. Start event begins when an user (a token in the Start Flag in Figure 1) programs the transceiver rate, and/or there is a change from the normal mode

3. After the synchronization, the analyser of the two ICs starts the analysis process, quantification, and bit-allocation of the actual speech frame, to the secure mode, or viceversa. Now the synchronisation task is started, and the IC that performs it becomes the master of the whole system. Syncho Frame (TX_SF process in Figure 1) is sent to slave system. During this process the IC slave answers with an Acknowledge Frame (TX_ACF) and, if no errors occurred during transmission (ACK_OK) both systems are programmed by changing the rate and/or enabling/disabling the cryptography. If the cryptosystem has been enabled, the noise generators are loaded with the key and both, the transmission and reception cryptosystems are thus synchronized. When it is available analyser asserts the SPEECH_FRAME_AVAILABLE flag (a token generated by the analyser), and the transmission of the Speech Frame (TX_VF) begins. This process is repeated until a new Start event is launched.

4. In parallel, when a speech Frame is received (RX_VF), if there is not transmission error, the process of speech synthesis is performed, and the synthesized speech is presented in the output of the Digital to Analog Converter. If a parity error has occurred, the synthesizer is not updated, thus repeating the former frame.

5. The Time_Out Flag (not visible in Figure 1) is asserted, if during the normal process of transceiving the normal communication is interrupted. This event occurs after a certain amount of time programmed by the user, measured in speech frame duration, and controlled by the watch-dog timer. After a Time_Out event, both subsystems sleep by dead-locking of the PN, achieved with the inhibition of the Enable_Synthesis conditions of the PN, and the Power-Down mode is activated. The initial state. At this level, we do not distinguish between a different kind of elements that form the system, because all of them are considered like processes, modelled with places in the PN. The places could represent either, a procedure for autocorrelation function estimation or the communication channel. The tokens in the PN represent, for example, speech samples from the analog to digital converter, hardware resources, the activation of a flag (a flip-flop), or a coded speech frame travelling through the channel. This independence and generalization of the system elements is a great advantage with several consequences:

1. Discrete-time ordering of the events occurring in the whole system are easy to manage: the designer can manage and schedule all processes in the system, using the required method, for instance using pipeline structures, exploiting parallelism, or a mix of both. The PN can be also easily modified and/or scaled.

2. The whole system is activated by the tokens flow: Independently of the tokens arrival time, the PN process only when a token is present. This is a great advantage because it is not necessary to work with absolute time but ordered events, and the consequences of this fact are: our systems at this level of modelling is IT; the system is also delay-independent and absolute time-independent thus giving us robustness: for example in a highly saturated network (like Internet) we do not know exactly when a frame will be received because the channel delay is non-stationary and unknown. The system will work correctly in any case of variable channel delay because the receiver section of the IA only will process a new frame when it is received.

3. Controlled dead-lock: we have called this characteristic proper of PNs sleep-lock. The controlled sleep-lock has been used to achieve the control of dynamic power consumption: if the flow of speech samples from the Analog to Digital Converter is locked, the analyser will not process more frames and for this reason the IA will not send them; the remote system after certain time will become timed-out and sleep-locked (Power Down mode), and at last the system which began the sleep-lock, also asserts the time-out condition activating the Power-Down mode. The fact of sleep-locking causes the reduction of the system dynamical activity, i.e. transitions in combinatorial logic, and change of state in the sequential one, affecting directly to the dynamical power.
Figure 1: Petri Net Model of the Codecrypt System: Two ICs Coupled Through a Noisy Channel.

We can also control the sleep-lock to achieve, in the CodeCrypt system, extra speech compression: a Voice Operated Keying system (VOX) manages the sleep-lock, processing and transmitting speech only when it is present (VOX switched on), during silences the system sleeps and no speech frame is transmitted (VOX switched off), this capability is called silence compression, very useful in saturated communications systems.

4. We can predict not only the correct behaviour of the system but also the ‘hidden states’ that can occur and which could produce instability or new useful ‘hidden functionality’: i.e. extrapolation capabilities.

Probability of error in the channel $P_{err}$ is modelled by using the stochastic properties of the PN. In general, we would be able to model a non-stationary channel, but for simplicity we will only illustrate the stationary case. A simple procedure to achieve the desired probability is the following:

1. The set of conditioner places are TX_SF, TX_VF and TX_ACF of both systems. The set of conditioned places are RX_SF, RX_VF and RX_ACF.
2. Probability of error of the channel is a fixed parameter, and using the total probability theorem we have the following relation:

$$P(error) = P_{err}$$

$$1 - P(error) = P(RxVF|TxVF) \cdot P(TxVF) +$$

$$P(RxSF|TxSF) \cdot P(TxSF) +$$

$$P(RxACF|TxACF) \cdot P(TxACF) = 1 - P_{err}$$

3. By assuming the following hypothesis, we can simplify highly the expression: If we observe the channel we will sample mainly speech frames, because synchronization occurs only during initialization of the system and rarely for change of rate or security mode. The other conditional probabilities belonging to the TX_VF column in the conditional probability matrix are assumed equally likely. With the other two columns, due to their low contribution to the total $P_{err}$, can be chosen freely: we have assigned the same distribution than the previous column, but giving the maximum probability to the ‘no error’ cases. Under these simple assumptions the former expression can be approximated as follows:

$$P(TxVF) = 1$$

$$1 - P_{err} = P(RxVF|TxVF)$$

Thus we can fix $P(RxVF | TxVF)$ and following the axioms
of the probability we can deduce the rest of the matrix:

\[
\begin{bmatrix}
P(\text{RxVF} \mid \text{TxVF}) & P(\text{RxVF} \mid \text{TxSF}) & P(\text{RxVF} \mid \text{TxACF}) \\
P(\text{RxSF} \mid \text{TxVF}) & P(\text{RxSF} \mid \text{TxSF}) & P(\text{RxSF} \mid \text{TxACF}) \\
P(\text{RxACF} \mid \text{TxVF}) & P(\text{RxACF} \mid \text{TxSF}) & P(\text{RxACF} \mid \text{TxACF})
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 - P_{\text{err}} & P_{\text{err}}/2 & P_{\text{err}}/2 \\
P_{\text{err}}/2 & 1 - P_{\text{err}} & P_{\text{err}}/2 \\
P_{\text{err}}/2 & P_{\text{err}}/2 & 1 - P_{\text{err}}
\end{bmatrix}
\]

Real statistic information could be fed-back from the real implemented system and backannotated for refinement purposes.

4. SPEECH CODING IC SIMULATION

The following example shows a typical scene of the CodeCrypt system. For speed purposes the remote system is emulated by a frame generator, reducing the time of simulation, for this reason only the local system can be the master system. The initial conditions of the simulation are given by a Reset sequence beginning at 0ms. And the following user generated events are programmed:

1. Programming of the Key Memory. The key of the address 6 is selected as the seed of the cryptosystem of the local system. The Power-Down mode with the Silence Compression (VOX) function is also activated. The Time Out is activated after the duration of two speech frames, i.e. the Watch-Dog Timer is loaded with a final count of two.
2. The user 1 launches the system synchronization activating the Secure mode (cryptography activated) in a time selected randomly and uniformly distributed between 0 and 125μs.
3. The channel probability error is \( P_{\text{err}} = 1\times 10^{-6} \).
4. A prepared speech segment with a silence is used in this example. It has been digitized with a 12-bits A/D converter using a sample frequency of 8KHz. In Figure 2.a we can see the sampled speech, and the VOX signal. Observe the delay in the disconnection of the VOX. This delay, with a minimum duration of a frame length (18ms), lets the analyser to codify the silence as the last frame sent before the system becomes sleep-locked, in this way the synthesizer of the remote system synthesizes this silence, thus avoiding the last speech frame being frozen until the activation of the VOX.

The simulation results are shown in Figure 3 and Figure 2.b. Figure 3.a shows the most relevant synchronization process of the two ICs in the form of high level timings:

After transmission of the SYNCHRO Frame by the local system, an ACK Frame sent by remote system is received, when this frame finishes, the analysis process starts, first in the remote system, followed by local system (this process is simultaneous when the channel delay is 0) and after 18ms, the duration of a speech frame analysis, the transmission of the speech frames begins in both systems. In Figure 3.b, it can be seen the effect of the VOX, during transmission: The channel is free approximately during the silence interval. In Figure 2.b, the synthesized speech signal is shown. We can observe how the silence has been synthesized and the delay respect to the original speech signal (Figure 2.a) in an amount of approximately 36ms: 18ms of frame analysis duration plus 18ms of frame transmission.

The duration of this simulation lasts 120 minutes running in a PC-1GHz. The same simulation performed with the VHDL system description in suited simulator lasted nine hours, running in a UltraSpare WS.

5. CONCLUSIONS

1. We have presented a formal description and simulation of a high complexity digital IC system. This modelling technique, based on stochastic PNs with time, let us to model the IC with independence of technology, and with any degree of depth we need, achieving highly accurate and fast simulation results.
2. Important properties of the PNs are used to achieve important topics in IC design: sleep-lock can be used to control dynamical power consumption. Tokens flow activation of the systems gives robustness, making the them independent of absolute time and thus independent of technology.
3. We have developed using this technique a real and highly complex digital system: CodeCrypt, making the use of high-level modelling tools an important and real framework for implementing digital ICs.

6. REFERENCES


Figure 2: Input Signals Used for the Simulation. 2.a. Shows the Original Sampled Signal, Quantified with 12 bits, 8kHz, and the Vox Signal. 2.b. Shows the Synthetic Signal Delayed around 36ms from the Original.

Figure 3: High level timings. (a) Shows Synchronization Process between the Local System and the Remote IC. (b) Normal Flow of Speech frames through the Channel: the Effect of the VOX can be seen in the Interruption of Communication During Speech Signal Silences.
CHEMICAL SIMULATION
INTEGRATION OF SIMULATION AND OPTIMIZATION FOR PRODUCTION SCHEDULING IN THE CHEMICAL INDUSTRY

Hans-Otto Günther and Gang Yang
Department of Production Management
Technical University of Berlin
Wilmersdorfer Str. 148
10585 Berlin, Germany
E-mail: ho.guenther@pm-berlin.net

KEYWORDS
Scheduling, simulation, chemical industry, linear programming, decision support system.

ABSTRACT

Both, simulation and optimization are considered as appropriate tools for production planning in the chemical industry. This paper outlines an approach for the integration of these different methodologies into one comprehensive decision support system, which consists of three major modules. The process/equipment module allows the user to interactively model the individual chemical processes and the available processing equipment. The planning module automatically generates the corresponding optimization model and provides different exact and heuristic solution methods. Finally, the visualization module converts the results from the optimization run into a Gantt chart representation and supports the automatic generation of a corresponding simulation model. The suggested approach has been tested using examples of chemical processes taken from the academic literature as well as a case study from industry.

INTRODUCTION

Today, many quantitative methodologies are available to support decision making in production planning and scheduling. Among these discrete event simulation and mathematical programming have received widespread acceptance in industry. Both methodologies have merits. For instance, discrete event simulation allows to analyze operations of production and logistics systems at a great level of detail and to consider dynamic variations which occur in the application environment, e.g. stochastic demand fluctuations or uncertainty of the chemical processing times. Although simulation supports the comparative evaluation of different production configurations or logistics control principles, simulation models do not provide analytical methods which are required to develop feasible or optimal production schedules. On the other hand, optimization models are necessarily based on simplifications of the production processes and on the aggregation of the production equipment. Typically, optimization models applied in industry are deterministic in nature. In particular, in supply chain management there is currently a strong trend to use purely deterministic planning approaches (cf. Günther and van Beek 2003). Moreover, most of the production scheduling models are based on a discrete representation of time and the assumption that material flows only take place at the beginning and the end of a period, while in simulation the occurrence of events is not restricted to the period boundaries.

![Diagram of decision support system]

Figure 1: Architecture of the decision support system

Hence, it seems appealing to combine simulation and optimization in order to gain advantages offered from both worlds (cf. Iassinovski et al. 2003). This paper outlines an approach for the integration of these different methodologies into one comprehensive decision support system. The approach focuses on application in the chemical industry. This type of industry shows a considerably increased complexity compared to discrete parts manufacturing (cf. Reklaitis et al.
1996). For instance, the complexity of scheduling chemical processes is determined by such factors as batch size constraints, shared intermediates, flexible proportions of input and output goods, production of by-products, limited predictability of processing times and yields, mixing and blending processes, carrying out processes without interruption, use of multi-purpose equipment, sequence and usage dependent cleaning operations, finite intermediate storage and use of product specific storage devices, cyclical material flows, no-wait production for certain types of products, usage of secondary resources, such as energy, steam, or cooling water, and complex packaging and filling operations. Very often, time and cost intensive cleaning procedures as well as the necessity of detailed quality control after each batch are the major motivations for operating the production system in campaign mode, i.e., the equipment units required by a particular type of product or process are set up according to the corresponding recipe.

The interactive decision support system suggested consists of three major modules, the first one for the definition of the individual chemical processes and the available processing equipment, the second one for the automatic generation and solution of optimization models, and the final one for the visualization of the optimization results. Figure 1 displays the architecture of the decision support system.

**PROCESS/EQUIPMENT MODULE**

The process/equipment module allows the user to interactively model the individual chemical processes and the available processing equipment. This module is based on a representation scheme, which reflects the real world entities from two different points of view, namely a process view and an equipment view. As a result, two different model representations are obtained. The equipment model represents the structure of equipment units employed to transform raw materials through intermediates to final products within a particular production network. The process model represents the production processes to be carried out and the interrelationships between them. In accordance with the required level of detail at different hierarchical levels, the equipment structure and the process structure can be depicted in a more or less detailed manner.

For representation of chemical processes the well-known task-state-network (STN) concept is used (cf. Kondili et al. 1993), which has found widespread acceptance in the chemical industry. A state-task-network consists of three types of elements: (1) state nodes, representing the feeds, intermediates, and final products, (2) task nodes, representing the processing steps, and (3) arcs, which indicate the material flow.

Simple drag and drop operations can be performed on the screen in order to define or modify the complete system of processing tasks and equipment units. Moreover, attributes are assigned to the individual objects on the screen, such as demand figures, processing times, batch sizes, or production and storage capacities. The related tables can be filled via spreadsheet tables or through manual input. Figures 2 and 3 depict the defined process structure and the corresponding production equipment for a sample process. For the implementation of the interactive decision support system the object-oriented simulation software eM-Plant (formerly Simple++) was used. Specifically, modelling tools provided by the process industry toolkit of eM-Plant/Simple++ were employed.

**Figure 2: Process view**

**Figure 3: Equipment view**
PLANNING MODULE

Based on the underlying representation of the chemical process and the available production equipment the planning module automatically generates the corresponding optimization model and provides different exact and heuristic solution methods. Hence, the planning module analyzes the various processing tasks and equipment units as well as the links between the objects, e.g. material flows and feasible task-equipment assignments which have been defined by the user. Also the attributes assigned to the various objects, e.g. batch sizes and feasible equipment units are considered.

From the many production planning and scheduling problems which occur in the chemical industry two specific problems have been selected for a prototype implementation within the interactive decision support system. The first one considers short-term scheduling of chemical batch processes, while the second one focuses on supply network planning in a multi-plant production setting.

Short-term scheduling of chemical batch processes

In multi-product, multi-purpose chemical batch plants the major scheduling task is the determination of a production schedule which satisfies given end-product requirements while taking batch size and various other constraints arising from the production process into account (cf. Kondili et al. 1993). Minimizing the makespan, i.e. completing the required production operations within the shortest possible time, is considered as the objective function. In the suggested decision-support system this type of planning problem is covered by a generic optimization model formulation which includes the various constraints of the planning problem in a general form. Constraints of this general model formulation refer to start and finish times of the various processing events, batch size limits, the feasible task-equipment assignments, stock balances and mass flow conditions, and storage capacities. For details the reader is referred to (Blömer and Günther 1998 and 2000).

The planning module generates an application-specific model by specifying the individual entities in the underlying generic model formulation. For instance, for a general class of processing equipment the individual equipment units are identified from the user-defined equipment view. At the same time, the various model parameters are identified, e.g. batch processing times or end-product demand. The application-specific model is then documented in a text file according to the formatting requirements of the AMPL algebraic modeling language. As an example, Figure 4 shows a single processing task which produces output products 7 and 9 from the two inputs 4 and 5 along with the automatically generated stock balances and batch size limits in AMPL notation. Using CPLEX as a powerful standard optimization software for linear and mixed-integer programming, the AMPL text file is read in and the actual optimization run is initiated. The output from the optimization run are the assignment of processing tasks to equipment units, the individual batch sizes, the sequence of batches and cleaning operations on each equipment unit, the detailed timing of the processing tasks as well as the distribution of material flows over time.

In the literature, problems of production planning and scheduling for chemical processes are most often formulated as a mixed-integer linear programming (MILP) model (see e.g. Kondili et al. 1993, Grunow et al. 2002 and 2003, Kallrath 2002). The computational burden, however, associated with optimally solving large-scale MILP models is prohibitive for most problems of realistic size. Therefore, efficient linear programming based scheduling heuristics have been developed which permit near-optimal solutions with to be obtained with only modest computational effort (cf. Blömer and Günther 1998 and 2000). These efficient LP-based heuristics may be called up in order to reduce computational times if necessary.

Stock balance

subject to STOCK {k in 1..S,t in 2..H}:
   j[k,t] = j[k,t-1]
   + sum {i in T_out[k], hh in C: not i in TC[hh]}
      rho_out[i,k] * sum {h in K[i]: t-T[h,i]>=1} m[h,i,t-T[h,i]]
   + sum {i in T_out[k]: card(C)==0}
      rho_out[i,k] * sum {h in K[i]: t-T[h,i]>=1} m[h,i,t-T[h,i]]
   - sum {i in T_in[k]} rho_in[i,k] * sum {h in K[i]} m[h,i,t]
   - D[k,t]
   + R[k,t];

Batch size limits

subject to BATCH_SIZE_MIN{u in U,i in TA[u],t in 1..H}:
   s[u,i,t] * A_min[u] <= m[u,i,t];

subject to BATCH_SIZE_MAX{u in U,i in TA[u],t in 1..H}:
   m[u,i,t] <= s[u,i,t] * A_max[u];

Figure 4: Processing task and corresponding modelling constraints
Supply network planning

Currently, there are strong trends in the chemical industry to co-ordinate the activities within the entire logistics network consisting of various production stages and plants at different locations and to introduce adequate modes of collaborative planning. Such supply networks call for a hierarchical structure of the planning and scheduling systems applied. Typically, production planning in a large production network can be sub-divided into three levels. Figure 5 illustrates the architecture of the corresponding hierarchical production planning system.

At the aggregate level, centralized supply network planning based on an aggregate view of equipment and processes is performed. At the intermediate level, detailed scheduling is carried out locally at each plant. This step involves, for instance, the determination of the sequence and number of batches and the assignment of equipment units to processing tasks. Accordingly, a more detailed view of the production equipment and processes applies. At the aggregate as well as the intermediate level, the continuous time based model formulation of (Grunow et al. 2003) is employed as a generic model formulation. Again an application-specific optimization model is generated from the user-defined process and equipment models. To solve the optimization model OPL studio as a powerful standard optimization tool is used (cf. Van Henderyck 1999). Finally, at the lowest level the individual production operations are initiated and controlled.

The results could be verified in a simulation model, which provides the necessary details of the real production environment. The proposed decision support system offers process and equipment views at different levels of aggregations and, therefore, supports hierarchical production planning for entire logistics networks. For the sake of brevity, these capabilities are not explicitly presented here.

VISUALIZATION MODULE

The visualization module analyzes the production schedule obtained by the planning module and supports the automatic generation of a corresponding simulation model. This feature, however, is restricted to short-term scheduling of chemical batch processes and the entities which have been defined in the initial process/equipment definition stage. Additional features may be added by the user in order to achieve the desired level of detail and to incorporate the particular behaviour of the production system. Such features may include, for instance, stochastic issues of the chemical processes and the equipment used, a more detailed representation of the various processes, and the consideration of additional resources, such as personnel and utilities.

The automatic generation of the simulation model comprises the following major steps. First, the underlying process-equipment structure and the related data are called up and linked to the simulation model. Second, from the corresponding short-term scheduling run of the optimization model, the number and characteristics of the generated batches, the sequence and timing of processing steps, the necessary cleaning operations and material transfers are obtained and converted into operations of the simulation model using pre-defined standard operations of the simulation software. Third, individual equipment units and recipes are modelled as components of the simulation model. Finally, a Gantt chart representation of the simulated schedule and other graphical representations of the simulation results are generated. All these steps are carried out automatically without any interaction by the user. Figure 6 shows an example of an automatically generated simulation model.

For the computer-based implementation of this module, again the process industry toolkit of eM-Plant/ Simple++ is used as a major software tool.

Figure 5: Architecture of the hierarchical production planning system
SUMMARY AND CONCLUSIONS

This paper has outlined the general architecture of an interactive decision support system for production planning and scheduling in the chemical industry. Special attention has been given to the integration of simulation and optimization modelling and the automatic generation of corresponding models. In spite of the huge effort required to build up individual simulation and optimization models the integrative approach promises considerable savings of development effort in a real application. However, a huge amount of software development is still required to create a professional planning and scheduling tool from the presented prototype decision support system.

ACKNOWLEDGEMENT

The authors gratefully appreciate the support from the German Academic Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under grant No. GU 459/3.

REFERENCES


AUTHORS BIOGRAPHY

HANS-otto Günther is Professor of Production Management at the Technical University Berlin. He received his PhD degree from the Free University of Berlin (Germany) and completed postdoctoral studies (Habilitat) at the University of Mannheim (Germany). His papers appeared in journals such as International Journal of Production Research, EJOR, OR Spectrum, Computers in Industry, Production Planning and Control, International Journal of Production Economics. Currently, he is Managing Editor of the journal “OR Spectrum: Quantitative Approaches in Management”. His research interests include Supply Chain Management and Advanced Planning Systems, operations planning and scheduling in the chemical industry, logistics control systems, and printed circuit board assembly.

GANG YANG has completed his doctoral studies in Industrial Management at the Technical University of Berlin. His research areas included the application of simulation and optimization methodology to supply network planning and detailed scheduling in the process industries. He is now working for an international consulting company.
A Tool Integration Framework for General Model-based Applications

G. Schopfer and W. Marquardt
Lehrstuhl für Prozesstechnik, RWTH Aachen, Turmstraße 46,
D-52056 Aachen, Germany
e-mail:{schopfer.marquardt}@lpt.rwth-aachen.de

May 21, 2004

Abstract

The software platform CHEOPS for tool integration in chemical process modelling was presented in [19]. Therein the application of steady state simulation to heterogeneous models was discussed. In the present paper a redesign of CHEOPS is introduced in order to allow the realization of general model-based applications. Specifically dynamic simulation and steady-state optimization are discussed. Parameter identification, formulated as an optimization problem, is also covered by the presented approach. The applicability of the proposed method to the case of parameter identification is illustrated in a case study of an ethylene glycol production process.

1 Introduction

Model-based simulation and optimization are extensively applied tools for the study of chemical processes. Given different design alternatives, process simulation is used to compare their performance with respect to e.g. their economics, energy consumption, environmental impacts or potential hazards. During the plant operation, simulation is further used as a basis for analyzing, improving and revamping the plant. On the other hand, if the design objective is formulated in terms of an economical objective, optimization can be employed.

Both simulation and optimization make use of a mathematical model of the process. One distinguishes between steady-state and dynamic models, describing the steady-state and the dynamic behavior of the plant, respectively. Correspondingly, we distinguish between steady-state and dynamic simulation and optimization [11].

As a special case of optimization, we consider parameter identification. This method is used to determine the unknown parameters of a process model-based on measurement data. The unknown model parameters are identified by minimizing the error between the given measurement data and the signal estimated by the model.

Due to the size and complexity of chemical processes, the resulting mathematical models are also large and complex. Thus, suitable software tools, called process modelling environments (PME), are used to support the engineer in developing and simulating these models [4].

Given the variety of existing PMEs, it is necessary to facilitate process model building by combining process unit models from different modelling tools to simulate an overall process model without recoding the process unit models in a single PME. This is referred to as model integration [2, 3]. In [18, 19] our approach to model integration for steady-state optimization, realized on the integration platform for chemical process modelling CHEOPS, was presented.

In the present paper our approach is extended towards the cases of dynamic simulation and steady-state optimization. The work on dynamic simulation is based on previous results [17]. The applicability of the method to parameter identification is illustrated in a case study of an ethylene glycol production process. Throughout the paper, the focus is set on presenting and explaining the functionality...
related to dynamic simulation and steady-state optimization. Our approach to tool integration and the underlying software architecture is described in detail in [19].

The paper is structured as follows. Section 2 summarizes the architecture of the realization of CHEOPS that is based on a neutral model representation. In Section 3 it is discussed how the neutral model representation is extended from steady-state to dynamic models. Section 4 first presents the general concept of model-based applications and discusses more specifically the realization of steady-state and dynamic simulation as well as steady-state optimization. Section 5 discusses the case study of a ethylene glycol production process. Finally, Section 6 summarizes the main advantages of the approach presented and points out directions for future research.

2 Overview over CHEOPS

Figure 1 gives an overview of the main runtime modules that are involved in a simulation or optimization with CHEOPS. The environment consists of a modelling tool server for each PME to be integrated and the CHEOPS client. The CHEOPS client communicates with the PMEs via a communication channel. It consists of the generic CHEOPS framework, and tool wrappers that integrate specific modelling tools into the CHEOPS framework. The CHEOPS framework can further be distinguished into the model-based application modules and the neutral model representation. The model-based application modules implement solution algorithms for simulation and optimization. They are based on the neutral model representation, which provides the process model in a format that is independent of the individual PMEs and supports different solution algorithms. A tool wrapper implements the communication between the CHEOPS client and a modelling tool server.

Different modelling tools provide different access mechanisms (COM/ CORBA interfaces, pipes, file-based, etc.) and run on different platforms with different operating systems (UNIX, Windows). In order to allow heterogeneous modelling with distributed modelling tools and in order to unify the access mechanism used in the different tool wrappers the CORBA object bus is chosen as communication channel. Consequently, the modelling tool servers have to provide access to the modelling tool via a CORBA interface. In the case that a modelling tool directly provides a CORBA interface, the modelling tool itself is the CORBA server. In other cases the modelling tool server consists of another, simple wrapper, which maps the modelling tool interface to the corresponding CORBA interface.

3 Neutral Process Model Representation

The conceptual object model of the neutral process model representation, as presented in more detail for steady-state models in [19], is shown in Figure 2. It is divided into the model structure representation and the mathematical representation. These two parts represent the structure of a process model, e.g. how several process unit models are connected to build a process model, and the model equations respectively. In the model structure representation a ModelContainer corresponds to one process unit model. The connection between the process units of a process model is represented by the Couplings between the Ports of the corresponding model containers. The mathematical representation provides the model equations in open and closed-formulation (represented by the classes OpenFormRepresentation and ClosedFormRepresentation) in order to support different solution algorithms for simulation [19]. Here the term open-formulation refers to a model formulation that provides full access to all model equations and provides the infor-
information that is required by an external numerical algorithm for the solution of a model (i.e. equation-residuals and jacobian matrices). On the other hand the term closed-formulation denotes a model formulation that combines a solution algorithm with the model equations. The closed-form representation does not provide access to the model equations but provides their solution, i.e. the model outputs as a function of the model inputs.

The parts of the neutral model representation that have been modified for extending it towards dynamic models are marked as grey boxes in Figure 2. The class OpenFormRepresentation was extended in order to also represent differential-algebraic equation (DAE) systems. Purely algebraic equation (AE) systems are simply treated as DAE systems without differential equations and without time dependent variables. Hence, the generalized class OpenFormRepresentation provides the information required by a Netwon type solver for the solution of a AE system and the information required by a numerical integrator for the solution of a DAE system.

For closed-form models the class ClosedFormRepresentation provides a common base class and provides common functionality. It is refined by the class ClosedFormAERepresentation for steady-state models and ClosedFormDAERepresentation for dynamic models (see Figure 2). These classes define additional methods for the evaluation of algebraic and dynamic models, respectively. The common base class for closed-form models allows to develop solution modules that only query this common base class (both for steady-state and dynamic models) and can be commonly used for steady-state and dynamic simulation (see Section 5).

Furthermore, to make the neutral model representation suitable for dynamic models, the concept of a variable had to be generalized (see Figure 3). Variable objects serve as storage of the numerical values of the model variables, e.g. they provide the storage for the inputs and the outputs of the closed-form representation. In a sequential modular simulation algorithm (see Section 4.1) the outputs of one closed-form representation object have to be propagated to the inputs of another closed-form representation object. This is done by copying the values between the corresponding variable objects. For algebraic models these variables contain one real number in case of a scalar quantity or a vector of real numbers in case of a vector quantity. For dynamic models a variable contains a vector of time dependent functions that are defined over a given time interval. This is reflected in the generalized object model for variables shown in Figure 3.

In the object model the abstract class Variable defines a unified interface for different variable implementations. This class is refined by the classes Vector and VectorFunction representing a vector of real numbers and a vector of continuous functions, respectively. For vector function variables different types of functions can be chosen (wavelet expansion, Legendre-, Laguerre-, or Hermite-polynomial). For time dependent variables of dynamic models the class VectorFunction represents a function of time that is defined over a certain time interval. The unified variable interface is very important to enable framework functionality that is common for steady-state and dynamic models.
4 Application Modules

We first introduce the general concept of applications and then discuss concretely the applications that have been realized in CHEOPS, namely steady-state and dynamic simulation and steady-state optimization.

4.1 General Concept

All application modules in CHEOPS have the same interface. This means that they are called for execution via the same commands including generic initialize and solve methods as well as the specification of the task to be solved. The concept of the general application interface is described by the object model shown in Figure 4.

Every ModelBasedApplication receives the problem specification in the form of a Flow-sheet object and an ExperimentSpecification. The experiment specification consists of two parts, the application specification and the value specification. The application specification defines which type of application and what numerical algorithm should be used. Furthermore it specifies relevant numerical parameters, e.g. numerical tolerances or a maximal allowed number of iteration steps. The value specification may consist of a single set of input values for a steady-state simulation, input profiles (defined over the time interval to be simulated) for a dynamic simulation or a set of measured data for a parameter identification. For the unified application interface it is important that all these types of variable specifications are represented by a common data format. The Flow-sheet object consists of the ModelContainer objects and the coupling objects that describe the process model. Furthermore, the flowsheet object provides further services to the applications on the level of the overall model. For example, it provides derivatives of the variable of one model container with respect to a variable of another model container. This is generally required for an optimization.

As shown in Figure 4 the class ModelBasedApplication is specialized by the different application modules, SteadyStateSimulation, DynamicSimulation and SteadyStateOptimization. For steady-state and dynamic simulation both sequential modular and simultaneous equation oriented solution algorithms are provided. The simultaneous equation oriented solution approach is based on one large equation system that models the whole process. This equation system consists of the equations of all process unit models and the so-called identity equations derived from the actual material/energy links among the process units [8]. For simulating the process model behavior a solution algorithm is applied to this overall equation system. In this solution approach the equations of the process unit models are obtained from the open-form representation objects of the model containers. The connection equations are obtained from the couplings. For steady-state simulation the Newton type solution algorithms NLEQ1S and NLEQ2 [13], and for dynamic simulation the DAE integration algorithm LIMEX [6] are integrated in CHEOPS.

The sequential modular solution algorithm requires the closed-form representation of all model containers. It successively executes the closed-form representation objects, propagating the variable values among the models according to the input-output connections described by the couplings. In case a recycle occurs, e.g. if there exists a process unit that has an output material stream fed back to one of its upstream units, the sequential modular solver offers a so called tearing/sequencing strategy, which tears a recycle stream and then gets it converged through iterative sequential calculations [5, 20].

Each of these applications queries the model containers of the flowsheet for the mathematical representation that is required by the numerical algorithm used by the application: The steady-state and dynamic equation oriented simulation modules receive the open-form representation, whereas the modules for steady-state and dynamic sequential modular simulation receive
the closed-form representation and the closed-form dynamic representation, respectively. The optimization module does not directly interact with the neutral model representation (see Section 4.3).

4.2 Steady-State and Dynamic Simulation

The design of the neutral process model representation allows to make common use of solution modules for steady-state and dynamic simulation as illustrated in Figure 5. Modules that make up the simulation applications are distinguished between individual solution modules for steady-state or dynamic, sequential modular or equation oriented simultaneous simulation and commonly used solution modules. In Figure 5 the parts of the neutral model representation that are used by the common solution modules are also shown.

For sequential modular simulation a module that implements an algorithm for the sequencing/tearing strategy is used for steady-state and dynamic simulation. It queries the model structure representation and the closed-form representation. Note, that both parts of the neutral model representation are common for steady-state and dynamic models.

For equation oriented simultaneous simulation a common solution module that provides services for equation system manipulation, such as the assignment of variables or the aggregation of equation systems, is used for steady-state and dynamic simulation. It is based on the model structure and the open-form representation.

4.3 Steady-State Optimization

Process optimization is based on minimizing some objective function that depends on the solution of the process model. In the case of parameter identification the goal is to determine some unknown parameters of the process model so that the prediction error of the model is minimal. The desired input-output mapping of a model is defined by a set of measurements of the process under investigation. If a least-squares criterion is used, the objective function to be minimized can be stated as follows:

\[
\Theta(p) = \sum_{i=1}^{n_{\text{exp}}} \sum_{j=1}^{n_{y}} \frac{1}{2} (y_{ij}(u^{m}, p) - y_{ij})^2
\]  

(1)

where \( p \) are the parameters, \( u^{m} \) are the measured process inputs, \( y^{m} \) are the measured process outputs and \( n_{\text{exp}} \) and \( n_{y} \) are the number of experiments and measured outputs, respectively. The model outputs \( y \) are calculated by solving the process model, given the measured process inputs \( u^{m} \) and parameters \( p \). Figure 6 shows the realization of optimization in CHEOPS. During the solution of an optimization problem the optimization module requires the module SteadyStateObjectiveFunction that provides the values of the objective function \( \Theta \). The objective function given by equation (1) is evaluated as a function of the parameters \( p \). In order to evaluate the values of \( \Theta \) the objective function has access to a steady-state simulation module. Since all model-based applications have the same interface, the solution algorithm that is most suitable for the simulation can be chosen without affecting the implementation of the objective function.

![Figure 6: Extension of CHEOPS towards optimization](image)
5 Case Study

We consider the steady-state modelling of an ethylene glycol production process. In this process ethylene oxide (EO) and water (H₂O) react to mono-, di-, tri- and tetra-ethylene glycol (MEG, DEG, TRIEG, TETEG). We assume that a steady-state model of the process is sufficient for a model-based investigation of the process behavior. The simplified process we consider here consists of three process units, a mixer, a reactor and a separation unit. In order to reduce the effort required for the modelling of the process, already existing models of process units should be reused.

In our scenario, the model for the mixer is available as a Fortran binary (developed originally for use in an inhouse Fortran based simulation tool). The separation unit is already developed in the modelling language Modelica [12] and is available as executable provided by the Modelica-based code generation tool UnitGen [7]. For the PA6 reactor no model is available and the general purpose modelling tool gPROMS [16] is used for modelling this process. The reactor mass and energy balance is formulated using the modelling language of gPROMS. The reaction kinetics can be described by the following four parallel reactions

\[
\begin{align*}
EO + H₂O & \underset{k₁}{\rightarrow} MEG \\
MEG + EO & \underset{k₂}{\rightarrow} DEG \\
DEG + EO & \underset{k₃}{\rightarrow} TRIEG \\
TRIEG + EO & \underset{k₄}{\rightarrow} TETEG
\end{align*}
\]

The reaction rate constants \( k_i \) can be calculated using the Arrhenius law

\[
k_i(T) = k_{0,i} \cdot \exp(-E/(R \cdot T)), \forall i, 1 \leq i \leq 4
\]

where the four constants \( k_{0,i}, i = 1, \ldots, 4 \), and the activation energy \( E \) are unknown. In order to determine these parameters, measurement data from the plant are available. In the plant, measurement sensors can only be installed at the inlet of the mixer and at the outlet of the separation unit. The rest of the plant is not accessible for measurements. In Figure 7 the measurement sensors are shown as lollipops.

The remaining task for the modelling of the process is the identification of the kinetic parameters. Since measurement data is only available at the inlet and outlet of the overall process, the parameter identification has to be done based on the overall model process. Due to the heterogeneity of the process model, CHEOPS is used for this task.

CHEOPS receives the problem specification via an XML input file. At runtime, a Python script parses the XML specification file, instantiates the neutral model representation, passes the problem specification to the selected application and runs the application. After the application has been completed, the results are written in an XML file for further use and are visualized in Excel tables and graphs.

Alternatively to creating an XML input file, a Graphical User Interface (GUI) can be used. The GUI allows a more convenient specification of the information and generates the XML input file.

The measurement data required for the case study was simulated using the following equations for the reaction constants \( k_i \)

\[
\begin{align*}
k₁ & = 60 \cdot N \cdot \exp(30.163 - (10583/(T))) \\
k₂ & = 2.1 \cdot k₁ \\
k₃ & = 2.2 \cdot k₁
\end{align*}
\]

with the normality \( N \)

\[
N = 0.0873
\]

as published in [15]. Furthermore, a constant temperature of \( T = 297K \) was assumed and the production of tetra-ethylene glycol was neglected. With this model 20 data points were generated and used for the identification of the model with unknown reaction constants \( k₁, k₂ \) and \( k₃ \). From prior knowledge the lower and upper bounds for these parameters could be set to 0.01 and 0.1, respectively. The initial values for the parameters were set to 0.5. The results of the parameter identification in CHEOPS and

![Figure 7: Process model of the Ethylene Glycol production process](image)

215
the values determined with the above formulas are given in the table below.

<table>
<thead>
<tr>
<th></th>
<th>$k_1$</th>
<th>$k_2$</th>
<th>$k_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>parker</td>
<td>0.02205</td>
<td>0.04632</td>
<td>0.04852</td>
</tr>
<tr>
<td>et al. [15]</td>
<td>0.02314</td>
<td>0.04641</td>
<td>0.04862</td>
</tr>
<tr>
<td>CHEOPS</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6 Discussion

The paper has presented the extension of our previous results on model integration for steady-state simulation to the cases of dynamic simulation and steady-state optimization in a unified framework. CHEOPS is designed with the objective of fulfilling the following requirements:

- Provide suitable solution algorithms,
- support different model-based applications,
- provide a software framework for further academic research, e.g. supporting the development and investigation of new solution algorithms.

For simulation two types of solution algorithms are realized in CHEOPS at the moment, the sequential modular and the equation oriented simultaneous solution algorithms. Due to their different numerical properties they have advantages for different process model structures. As an example, in the process modelling tool AspenPlus [1] both solution algorithms are provided for steady-state simulation [10]. On the other hand, for dynamic simulation commercial tools commonly use the simultaneous equation oriented approach to simulation [16], as it is being considered to be the most versatile one [14]. However, in the context of tool integration, there are cases where the sequential modular approach is the only feasible solution for dynamic simulation [9]. To accommodate these cases, CHEOPS provides a sequential modular solution algorithm as well for dynamic process models.

CHEOPS is designed in a way that allows the extension towards new model-based applications without affecting the existing implementation. This is enabled by the unified interface for applications and a neutral model representation that comprises a unified structure representation and an extendable mathematical representation. A new software module realizing a new model-based application interacts with its environment as all other application modules. In case it requires another mathematical model formulation, the mathematical representation can be extended accordingly.

The software design further allows the reuse of solution modules by using them as building stones in different contexts. This was illustrated in Figure 5 for steady-state and dynamic simulation and in Figure 6 for optimization. Consequently, new algorithms can be easily realized and investigated within the framework.

Future work comprises the development of advanced solution algorithms within CHEOPS and its further application to the simulation of crystallization processes [9] and hybrid (mixed-empirical-physical) modelling.

7 Acknowledgements

This work has been supported in part by DFG (Deutsche Forschungsgemeinschaft) in the collaborative research center IMPROVE (CRC 476) and by the European Union in the project Global CAPE-OPEN. The authors thank their colleagues of CRC 476, and in particular L. von Wedel, J. Wyes, S. Kulikov and R. Schneider for many fruitful discussions.

References


Network Security Process Modelling applied to Cross-organizational Use of Networked Information Resources in E-commerce environment

Gustavo Santana

1 Mexican Petroleum Institute (IMP), Mexico

Abstract. In this paper the most relevant works related to Network Security Process are reviewed. Network Security Process modelling tries to capture the main characteristics of the set of activities performed to obtain a Network Security Environment, and a variety of models have been created for this purpose. A process model can be used to define a recommended Network Security Process. This paper deals with the prescriptive Network Security Process modelling and the most relevant results of this approach are presented. Moreover this paper presents our exploration into an Organization Ontology for the Authentication and Access Management Security Enterprise Model. Its primary focus has been in linking structure and behavior through the concept of security empowerment. Security empowerment is the right of an organization agent to perform status changing actions. This linkage is critical to the unification of enterprise models and their executability.

1. Introduction

Enterprises are increasingly dependent on their information systems to support their business activities. Compromise of these systems either in terms of loss or inaccuracy of information or competitors gaining access to it can be extremely costly to the enterprise.

What is an organization and how do we model it in a security information system? Many disciplines have explored the former and every information system built has created a version of the latter. The purpose of this paper is to explore the latter from the perspective of Artificial Intelligence. As authentication and access management control in information systems play a more active role in the management and operations of an enterprise, the demands on these systems have also increased. As institutions implement networked information strategies which call for sharing and licensing access to information resources in the networked environment, authentication and access management have emerged as major issues which threaten to impede progress. While considerable work has been done over the last two decades on authentication within institutions and, more recently, in support of consumer-oriented electronic commerce on the Internet, a series of new technical and policy issues emerge in the cross-organizational authentication and access management context. This paper is intended to serve several purposes:

- To identify and scope the new issues that emerges in the cross-organizational setting and to provide a framework for analyzing them.
- To map out the various best-practice approaches to solving these problems using existing and emerging technology so that institutions and information providers can make informed choices among the alternatives and consider how these choices relate to institutional authentication and access management strategies.
- To provide a common vocabulary and framework to assist in the development of licensing and resource-sharing agreements, and to highlight technical and policy considerations that need to be addressed as part of these business negotiations.
- To lay the foundation for possible follow-on formal or de facto community standards development in access management. If large scale use of networked information resources is to flourish, we need to move away from the specialized case-by-case access management systems in use today and towards a small number of general approaches which will let institutionally-based access management infrastructures interoperate with arbitrary resources.

Departing from their traditional role as simple repositories of data, information systems must now provide more sophisticated support to manual and automated decision making; they must not only answer
queries with what is explicitly represented in their Security Enterprise Model, but must be able to answer queries with what is implied by the model. The goal of the Authentication and Access Management Security Enterprise Modeling is to create the next generation of Security Enterprise Model, a Common Sense/Criteria Enterprise Model. By common sense/criteria we mean that a Security Enterprise Model has the ability to deduce answers to queries that require relatively shallow knowledge of the domain. We are taking what can be viewed as a “second generation knowledge engineering” approach to constructing our Common Sense/Criteria Security Enterprise Model, rather than extracting rules from experts, we are “engineering ontologies.” An ontology is a formal description of entities and their properties, relationships, constraints, behaviors. Our approach to engineering ontologies begins with defining ontology’s requirements; this is in the form of questions that ontology must be able to answer. We call this the competency of the ontology. The second step is to define the terminology of the ontology – its objects, attributes, and relations. The third step is to specify the definitions and constraints on the terminology, where possible. The specifications are represented in First Order Logic and implemented in High Level Functional Programming Language. Much of our effort has been in creating representations of organization behavior: activity, state, causality and time, and the objects they manipulate: resources [Fadel 94], authorization, authentication, access control management.

This paper describes the organization ontology being developed as part of the Authentication and Access Management Security Enterprise Model as part of the Mexican Oil Institute’s Information Security Project. In particular it focuses on organization structure, roles, authority and security empowerment.

2. Security Threats in Cross-organizational Use of Networked Information Resources

A secure system is a system that provides a number of services to a selected group of users and restricts the ways those services can be used. A security service is a software or hardware layer that exports a safe interface out of an unprotected and possibly dangerous primitive service. In order to build a security service we need a security architecture. Security risks exist throughout the agent life-cycle. These risks are present during agent management, registration, execution, agent-to-agent communication, user-agent interaction, and agent mobility. In a heterogeneous environment security becomes an extremely sensitive issue. We identify the threats that exist in an agent-based infrastructure. We can have: misuse of execution environment by mobile agents, misuse of agents by other agents, misuse of agents by the execution environment, misuse by the underlying network infrastructure. Mainly all security efforts target the first category and a big part of the second one. But misuse of agents by the host isn’t touched almost at all. The following six security threats apply to agents and multi-agent systems:

Disclosure: A breach in the confidentiality of an agent's private data or meta-data. For example, an entity eavesdrops on the communication between agents and extracts information on the goals, plans, capabilities, etc. of these agents.

Alteration: The unauthorized modification or corruption of an agent, its state, or data. For example, an Agent Communication Channel modifies the content of a message.

Copy and Replay: An attempt to copy an agent, or a message, and clone or retransmit it. For example, a malicious platform creates an illegal copy, or a clone, of an agent, or a message from an agent is illegally copied and retransmitted.

Denial of Service: An attack that attempts to deny resources to the platform or an agent. For example, an agent floods the Directory Facilitator Agent with requests and the Directory Facilitator is unable to provide its services to other agents.

Repudiation: An agent, or agent platform, denies that it has received/sent a message or taken a specific action. For example, a commitment between two agents as the result of a contract negotiation is later ignored by one of the agents, the agent denies the negotiation has ever taken place and refuses to honor its part of the commitment.

Spoofing and Masquerading: An unauthorized agent, or agent platform, claims the identity of another, authorized or unauthorized, agent or agent platform. For example, an agent registers as a Directory Facilitator Agent and therefore receives information from other registering agents.

In order to handle these threats, the following security services should be implemented:

Confidentiality. The message is encrypted for the receiver. An eavesdropper won’t be able to view or derive the original plain text.

Integrity. A one-way hash of the message (i.e., message digest) is calculated and sent along with the message to the receiver. The recipient is then able to verify whether the message has been tampered with.
**Authentication.** The message is signed on behalf of the sender, the recipient is able to verify the sender’s identity.

**Non-repudiation.** The agent requests a level of authentication, integrity, and confidentiality to ensure non-repudiation and receives verifying information.

### 3. Credentials and Authentication

Because agents are programs, they are intangible and live in a virtual world, we connect the trust model of such an infrastructure with the trust model of real world in order to make security critical decisions. That basically means that since every agent acts on behalf of a user or generally an entity we check to see if we trust that entity and indirectly trust the agent. The connection between those two worlds, the virtual one of agents and the real one is done via the digital certificates. A digital certificate is an object (file or message) signed by a certification authority that certifies the value of a person's public key. X509 [3] certificates of the International Standard Organization are the most popular, so we also adopt them in our design.

An agent is signed by one or more entities. Those entities can be either the creator of the code, the user that dispatched the agent (usually this is also the creator), a place of a host and generally any entity that holds a valid certificate. Signing an agent guarantees that i) the creator is the one claimed by the agent, ii) agent's code (at least the signed part) has not been tampered by a third party during transportation. Signing doesn’t guarantee that the agent will execute correctly (safety). Furthermore one place can encrypt the agent with the public key of the destination place (only the destination place has the private key to decrypt the agent), protecting in this way the agent while it traverses the net until it reaches the final destination. In order to ensure secure external communication we don’t use any homedown solutions but instead we use the SSL (Secure Socket Layer) protocol. TLS standard (Transport Layer Security) is also another option.

Credentials also touch indirectly the “malicious host” problem. Since each place (or at least each agency) has its own certificate there is proof that this agent is mapped to a legal user who bears responsibility of the behavior of the agency. An agent (based on a trusted host) prior to transportation can get next host's credential and decide whether to migrate and what to compute on the specific host. Furthermore it can ask the place to sign the results with its private key, so it can prove that those results were obtained during the execution on that specific place (repudiation problem). Non-changing parts of the agent should be signed for maximum protection.

### 3.1 Means of authentication and authorization

In this section the actual methods or processes that are used to authenticate the identities of users are discussed. The authorisation of the user to gain access to services or resources can be carried out in a system after the user has been authenticated and his identity is resolved through the use of access control lists (ACL), to determine what the particular user is authorised to do. Authorisation is thus at maximum as accurate and correct as the process of authentication. A mechanism like the SPKI could be used, to avoid authentication of the user, but to still provide a reliable authorisation.

Some mechanism for implementing mobile authentication and authorization are:

- **Passwords.** Passwords associated to user names (something that the person knows) are a simple way of authentication. There are several authentication schemes that make use of passwords in combination with some other factor. A simple extension of passwords is one-time passwords.

- **Password with a token.** Passwords can be used in combination with some physical object (something that the person owns). This concept has been extended with the use of integrated circuit cards (ICC) or smart card. A challenge and response method is used in authenticating the user. "Synchronous one-time passwords" [5] is another similar technique.

- **Biometrics.** Biometrics authentication techniques include fingerprint recognition, retinal scanning, hand geometry scanning and handwriting and voice recognition [5]. These techniques are all based on the physical properties of a person (something he owns / is).

- **Digital Signatures.** When a PKI is put in place, digital signatures can be used to authenticate users. The following sequence of actions has to be carried out in order to authenticate a user by his digital signature:
  1. The user requests access to the service or system.
  2. The system generates some data for the user to encrypt using his private key. Then the data is sent over to the user.
  3. The user concatenates the data received from the system and a time stamp and encrypts the whole sequence. (N.B. It is a good practice that e.g. a time stamp is concatenated to the
data, so that the data to be encrypted cannot completely be decided by some untrusted party. This is to avoid the possibility of a ‘Chosen plain-text attack’ as described in [12].) Then the encrypted data (the cipher text) is sent back to the system. Along with the encrypted data a link to the certificate (or the certificate itself) of the user is sent.

4. The system decrypts the received information with the public key of the user, found in the certificate.

5. The system verifies that the decrypted information is composed of the originally generated data and a valid timestamp. If this seems to be OK, the system has successfully authenticated the user.

4. Agent Platform

4.1 Defining The Cross-Organizational Access Management Problem

The basic cross-organizational access management problem is exemplified by most licensing agreements for networked information resources today; it also arises in situations where institutions agree to share limited-access resources with other institutions as part of consortia or other resource sharing collaborations. In such an agreement, an institution -- a university, a school, a public library, a corporation -- defines a user community which has access to some network resource. This community is typically large, numbering perhaps in the tens of thousands of individuals, and membership may be volatile over time, reflecting for example the characteristics of a student body. The operator of the network resource, which may be a web site, or a resource reached by other protocols such as Telnet terminal emulation or the Z39.50 information retrieval protocol needs to decide whether users seeking access to the resource are actually members of the user community that the licensee institution defined as part of the license agreement. Note that the issue here is not how the licensee defines the user community -- for example how a university might define students, staff members and faculty (all of the problems about alumni, part time and extension students, adjunct faculty, affiliated medical staff and the like); it is assumed that the institution and the resource operator have reached some satisfactory resolution on this question. Rather, the issue is one of testing or verifying that individuals are really a member of this community according to pre-agreed criteria, of having the institution vouch for or credential the individuals in some way that the resource operator can understand. Such arrangements are often called “site” licenses, but this term is really inaccurate; while physical presence at a specific site may be one criteria for having access, a better term is “group” license or “community” license, emphasizing that the key consideration is membership in some community, and that physical location is often not the key membership criteria.

Progress in inter-organizational access management will benefit everyone. To the extent that resource operators and licensing institutions can agree on common methods for performing this authentication and access management function, it greatly facilitates both licensing and resource sharing by making it quick, easy and inexpensive to implement business arrangements. It benefits users by making their navigation through a network of resources provided by different operators more seamless and less cumbersome. The central challenge of cross-institutional access management is not to set up barriers to access; it is to facilitate access in a responsible fashion, recognizing the needs of all parties involved in the access arrangements.

While this paper will give some particular emphasis to issues that arise in the higher education and library communities (particularly at the policy level) the problem under consideration here is very general, and in fact occurs in general corporate licensing of networked information services, or cooperation among business partners.

Note that there are several other cross-organizational authentication, authorization and access management issues which are beyond the scope of this paper, including the authentication of service providers and verifying the integrity and provenance of information retrieved from networked resources.

4.2 Authentication, Authorization, and Access Management Definitions

Throughout the rest of this paper we’ll use the general terms “resource operator” to cover publishers, web site operators, and other content providers (including libraries and universities in their roles as providers of content), and “licensee institution” to cover organizations such as universities or public libraries that arrange for access to resources on behalf of their user communities.

Authentication and authorization actually have very specific meanings, though the two processes are often
confounded, and in practice are often not clearly distinguished. We will use the term “access management” to describe broader systems that may make use of both authentication and authorization services in order to control use of a networked resource.

Authentication is the process where a network user establishes a right to an identity -- in essence, the right to use a name. There are a large number of techniques that may be used to authenticate a user -- passwords, biometric techniques, smart cards, certificates. Note that names need not correspond to the usual names of individuals in the physical world. A user may have the rights to use more than one name: we view this as a central philosophical assumption in the cross-organizational environment. There is a scope or authority problem associated with names; in essence, when a user is authorized to use an identity this is a statement that some organization has accepted the user’s right to that name. For authorization within an institution this issue often isn’t important, and in some schemes a user may only have a single identity; for cross-organizational applications such as those of interest here, this relativistic character of identity is of critical importance. A user may have rights to use identities established by multiple organizations (such as universities and scholarly societies) and more than one identity may figure in an access management decision. Users may have to decide what identity to present to a resource: they may have access because they are a member of a specific university’s community, or a member of a specific scholarly society, for example. Making these choices will be a considerable burden on users, much like trying to shop for the best discount rate on a service that offers varying discounts to different membership and affinity groups (corporate rate, senior citizen rate, weekly rate, government rate, etc.).

A name or identity has attributes associated with it. These may be demographic in nature -- for example, this identity signifying a faculty member in engineering, or signifying a student enrolled in a specific course -- or they may capture permissions to use resources. Attributes may be bound closely to a name (for example, in a certificate payload) or they may be stored in a directory or other demographic database under a key corresponding to the name. Attributes may change over time; for example, from semester to semester the set of courses that a given identity is associated with may well change. Just because some system on a network has knowledge of a name does not necessarily imply that it has access to attributes associated with that name. There is a fine line between rights to names (authentication) and attributes; for some purposes, simply knowing that a user has a right to a name from a given authorizing authority may itself represent sufficient information (an implicit attribute, if one wishes) that can support access management decisions.

Authorization is the process of determining whether an identity (plus a set of attributes associated with that identity) is permitted to perform some action, such as accessing a resource. Note that permission to perform an action does not guarantee that the action can be performed; for example, a common practice in cross-organizational licensing is to further limit access to a maximum number of concurrent users from among an authorized user community.

Note that authentication and authorization decisions can be made at different points, by different organizations.

Some libraries are establishing consortia which involve reciprocal borrowing and user-initiated interlibrary loan services; in a real sense these consortia are developing what amounts to a union or distributed shared patron file. One can view this as moving beyond just common authentication and access management to a system of shared access to a common directory structure for user attributes, and a common definition of user attributes among the consortium members. This is an example of a situation where very rich attributes are available to each participant in the consortium as they make authorization decisions; interlibrary loan and reciprocal borrowing represent a much richer and more nuanced set of actions than would be typical of a networked information resource.

In order for the security system to work, almost all the elements of the agent platform (AP) have to intervene to make a secure agent platform, in this section we review the participation of each of these elements (fig. 1).
4.4 Policy Manager

The Policy Manager is responsible for managing the policy schemes stored in the policy database. By separating the policy DB from the enforcement engine we insert a dynamic way of policy modification. The security policy defines the access each agent has to resources. Signed agents can run with different privileges based on the identity of the person who signed it. Thus users can tune their trade-off between security and functionality (of course within limits given by administrator).

When an agent comes to an AP then he is subjected first to the general platform's policy which is set by the platform administrator. Subsequently after passing successfully that control the agent is subjected to the service's specific policies. It is clear that with this sequential check of policies we avoid the problem of granting contradictory access rights for the same action by different policies. The policy of the platform is always first checked and therefore it has precedence over service's policy. This architecture makes it easy for an enterprise to set-up a platform and then provides advanced services to its customers.

Notification of malicious agents that have attacked other hosts can be distributed in the network. When the AP receives such a notification it can add a line to platform’s general policy (that is always checked first) that will not allow agents that bear those malicious characteristics to execute.

So we have role-based policy, group policy, clearance labels, domains etc. Furthermore by grouping policies we allow for faster execution times while trying to enforce the policy. In our system all security checks are identity-based in order for an agent to enter a platform or request a service from other agents. After an agent successfully enters an AP, future security checks become role-based. Thus we don’t have each time to verify agent's credentials. We check only to see in which platform the agent resides and what the appropriate policy for that platform is.

4.4.1 General Security Policy

It is the IT security policy of the policy manager that:

1. Each agency shall operate in a manner consistent with the maintenance of a shared, trusted environment within resource and component manager for the protection of sensitive data and business transactions. Agencies may establish certain autonomous applications, including those hosted by an Applications Service Provider or other third party, outside of the shared, trusted environment, PROVIDED the establishment and operation of such applications follows all guidelines as set forth in this security policy and does not jeopardize the enterprise security environment, specifically:
● The security protocols (including means of authentication and authorization) relied upon by others; and
● The integrity, reliability and predictability of the Organisational backbone network.

2. Each agency shall establish its secure state business applications within the guidelines of the Policy Manager and resource manager Network Infrastructure. This requires that all parties interact with agencies through a common security architecture and authentication process. Enforcement Engine shall maintain and operate the shared infrastructure necessary to support applications and data within a trusted environment.

3. Furthermore, each agency that operates its applications and networks within the whole Organisational Network Infrastructure must subscribe to the following principles of shared security:
   ● Agencies shall follow security standards established for selecting appropriate assurance levels for specific application or data access and implement the protections and controls specified by the appropriate assurance levels;
   ● Agencies shall recognize and support the state’s standard means of authenticating external parties needing access to sensitive information and applications;
   ● Agencies shall follow security standards established for securing servers and data associated with the secure application; and
   ● Agencies shall follow security standards established for creating secure sessions for application access.

4. Each agency must address the effect of using the Inter-networking protocols to conduct transactions for state business with others. Plans for Internet-based transactional applications, including but not limited to e-commerce, must be prepared and incorporated into the agency's portfolio and submitted for security validation.

5. Each agency must review its Information transactions security processes, procedures, and practices at least annually and make appropriate updates after any significant change to its business, computing, or telecommunications environment. Examples of these changes include modifications to physical facility, computer hardware or software, telecommunications hardware or software, telecommunications networks, application systems, organization, or budget. Practices will include appropriate mechanisms for receiving, documenting, and responding to security issues identified by third parties.

6. Each agency must conduct an IT Security Policy and Standards Compliance Audit once as frequently as possible. The audit must be performed by knowledgeable parties independent of the agency’s IT organization, such as the General Agent Auditor. The work shall follow audit standards developed and published by the General Agent Auditor.

4.5 Cryptographic Agent

It was decided that cryptographic mechanisms should not be built into the APMS by itself, but rather be modeled as a service agent. This agent (CryptoAgent) would be activated by the APMS whenever encryption or decryption was needed. The advantage of this modular design was that other components (even mobile agents) could use the functionality offered by CryptoAgent. Furthermore, the platform could be made unaware of any additions or changes to cryptographic functions offered by this agent.

5. Ontology Competence

A problem in the engineering of ontologies is their evaluation. A number of criteria have been proposed including [Fox et al. 93] [Gruber 93]:

● Functional Completeness: Can the ontology represent the information necessary to support some task?
● Generality: To what degree is the ontology shared between diverse activities such as engineering design and production, or design and marketing? Is the ontology specific to a sector, such as manufacturing, or applicable to other sectors, such as retailing, finance, etc.?
● Efficiency: Does the ontology support efficient reasoning, i.e., space and time, or does it require some type of transformation?
● Perspicuity: Is the ontology easily understood by the users so that it can be consistently applied and interpreted across the enterprise? Does the representation “document itself?”
● Precision/Granularity: Is there a core set of ontological primitives that are partitionable or
do they overlap in meaning? Does the representation support reasoning at various levels of abstraction and detail?

- **Minimality:** Does the ontology contain the minimum number of objects (i.e., terms or vocabulary) necessary [Gruber 93]?

The criterion we have found most useful is *competence*. For any task in which the ontology is to be employed, the task imposes a set of requirements on the ontology. These requirements can best be specified as a set of queries that the ontology should be able to answer, if it contains the relevant information. These requirements, which we call competency questions, are the basis for a rigorous characterization of the information that the ontology is able to provide to the task. Competency questions are benchmarks in the sense that the ontology is necessary and sufficient to represent the tasks specified by the competency questions and their solution. They are also those tasks for which the ontology finds all and only the correct solutions. Tasks such as these can serve to drive the development of new ontologies and also to justify and characterize the capabilities of existing ontologies.

This characterization of competency raises an important issue: where does the representation end and inference begin? If no inference capability is to be assumed, then query processing is reducible to “looking up” an answer that is represented explicitly. In contrast, object/semantic network representations assume at least inheritance as a deduction mechanism. In defining an ontology a key question then becomes: should we be restricted to just a terminology? Should the terminology assume an inheritance mechanism, or some type of theorem proving capability as provided, say, in a logic programming language with axioms restricted to Horn clauses (i.e., Prolog)? What is the *deductive capability* that is to be assumed by an ontology? In the Authentication and Access Management Security Enterprise Modeling we assume a theorem prover of the power of High Level Functional Programming Language.

The basic entities in the Authentication and Access Management Security ontology are represented as objects with specific properties and relations. Objects are structured into taxonomies and the definitions of objects, attributes and relations are specified in first-order logic. An ontology is defined in the following way. We first identify the objects in our domain of discourse; these will be represented by constants and variables in our language. We then identify the properties of these objects and the relations that exist over these objects; these will be represented by predicates in our language. We next define a set of axioms in first-order logic to represent the constraints over the objects and predicates in the ontology.

This set of axioms provides a declarative specification for the various definitions and constraints on the terminology. Further, we need to prove the competency of the ontology. The ontology must contain a necessary and sufficient set of axioms to represent and solve these questions, thus providing a declarative semantics for the system. It is in this sense that we can claim to have a competent ontology, and it is this rigor that is lacking in previous approaches to ontology engineering.

The competency questions are generated by requiring that the ontologies be necessary and sufficient to support the various tasks in which it is employed. Within our applications of activity/time ontology, these include:

- Planning and scheduling -- what sequence of activities must be completed to achieve some goal? At what times must these activities be initiated and terminated?
- Temporal projection -- Given a set of actions that occur at different points in the future, what are the properties of resources and activities at arbitrary points in time? This includes the management of resources and activity-based costing (where we are assigning costs to resources and activities).
- Execution monitoring and external events -- What are the effects on the enterprise model of the occurrence of external and unexpected events (such as machine breakdown or the unavailability of resources)?
- Hypothetical reasoning -- what will happen if we move one task ahead of schedule and another task behind schedule? What are the effects on orders if we buy another machine?
- Time-based competition -- we want to design an enterprise that minimizes the cycle time for a product [Blackburn 91]. This is essentially the task of finding a minimum duration plan that minimizes action occurrence and maximizes concurrency of activities.

We propose the following set of competency questions for the organization ontology (We do not view these competency questions as being complete but indicative of what is needed.).

**Structure Competency**

- What role does an agent play?
- Which division does the agent belong to?
- Who must the agent communicate with?
- What kinds of information does the agent communicate?
• Who does the agent report to?
• Is a role a generalization of another role?

In linking the structure of an organization with the behavior of agents within the organization, we must define how the organization ontology is integrated with the activity ontology. If we consider an organization to be a set of constraints on the activities performed by agents, then the competency questions for the organization ontology are extensions of the temporal projection and plan existence problems to incorporate the abilities and obligations of agents. The temporal projection problem is used to characterize the constraints that agents must satisfy to be able to perform activities, and plan existence characterizes the set of achievable goals.

**Behavior Competency**
- What are the goals of the organization?
- What are the goals of a role?
- What are the goals of person X?
- What activities are available for a role to achieve its goal?
- What resources are available to achieve a goal?

**Authority, Security empowerment and Commitment Competency**
- What resources does the person have authority to assign?
- In order to perform a particular activity, whose permission is needed?
- What activities may a person execute under their own authority?

**Conclusions**

Ontologies are shared views of domains. They provide conceptualizations that are agreed upon by participants in collaborative action and decision making. The explicit existence of such shared perspectives makes it possible for both people and programs to collaborate by ensuring that everybody makes the same distinctions and uses the same terms with the same meaning. The paper presents our preliminary exploration into an organization ontology for the Authentication and Access Management Security enterprise model. The ontology views organizations as composed of agents playing roles in which they are acting to achieve specific goals according to various constraints defining the “rules of the game”. A primary focus has been in linking structure and behavior through the concept of security empowerment.

Security empowerment is the right of an organization agent to perform status changing actions. This linkage is critical to the unification of enterprise models and their executability. Further work can be done on completing the axiomatization of the ontology and extending it to capture other concepts such as skill, intention, access right to the information system of the organization, etc.

Both proxies and credential-based authentication schemes seem to be viable approaches. Proxies have the advantage of compartmentalizing and modularizing authentication issues within an institution. But they also place heavy responsibilities upon the licensee institution to operate proxy servers professionally and responsibly. Proxy servers will become a focal point for policy debates about privacy, accountability and the collection of management information; successful operation of a proxy server implies that the user community is prepared to trust the licensee institution to behave responsibly and to respect privacy. Similarly, resource operators have to trust the licensee institution to competently implement and operate a local authentication system; anomaly monitoring of aggregated traffic from a proxy server by a resource operator is very difficult, and the resource operator will have to largely rely on the institution to carry out a program of anomalous access monitoring.

A cross-organizational authentication system based on a credential approach has the advantage of greater transparency. Resource operators can have a higher level of confidence in the access management mechanisms, and a much greater ability to monitor anomalous access patterns. The downside is much greater complexity; issues of privacy, accountability and the collection of management statistics become a matter for discussion among a larger group of parties. Further, it seems that a credential system means that there has to be cross organizational interdependency in order to avoid systemic compromise of the authentication system, as opposed to a simple relationship of trust -- recognized in a contract -- for the proxy approach.

One point that seems clear is that an institutional public key infrastructure may not extend directly to a cross-institutional one; it may be desirable to issue community members a set of pseudonymous certificates for presentation outside the institution as well as individually identified ones that are used within the institution in order to provide a privacy firewall while still maintaining some level of accountability.

IP source filtering does not seem to be a viable general solution, although it may be very useful for some niche applications, such as supporting public workstations or kiosks. It can be used more widely -- indeed today it usually is the basic access management tool -- but it definitely cannot support remote users flexibly in its basic form. Most real-world access management systems are
going to have to employ multiple approaches, and IP source address filtering is likely to be one of them.

A final issue: this paper has focused on inter-institutional issues in authentication and access management. It should be clear that the role of the licensee institution as a mediator adds some very significant value for the members of the user community. There are many users of networked information resources who do not have a natural affiliation with a licensee organization, and who thus do not have a way to obtain these benefits. We can expect these users to seek affiliations -- such as that of alumni -- which allow them to obtain these benefits. The idea of being able to have a single ID that allows access to a vast array of networked information resources is a very powerful one, and it is one that today is available only in an institutional context.

References


SIMULATION IN ENERGY
POWER PLANT SIMULATION
OBJECT-ORIENTED LIBRARY FOR THERMAL POWER PLANT SIMULATION

Francesco Casella and Alberto Levi
Dipartimento di Elettronica e Informazione
Politecnico di Milano
Piazza Leonardo da Vinci, 32
20133 Milano
Italy
E-mail: {casella,leva}@elet.polimi.it

KEYWORDS

ABSTRACT
This paper presents a library for the simulation of thermo-hydraulic power generation plants. The library is written in the Modelica language, and exploits the object-oriented paradigm in a way that is particularly suitable for this specific domain. After introducing and motivating the modelling principles adopted, an overview of the library (that is available to the scientific community via the web) is given, and a short example of its validation is presented.

INTRODUCTION
In the design of modern thermal power generation plants, dynamic simulation plays a crucial role, in particular when the control design is addressed. Appropriate methods and tools are required, as the rapid evolution both of the generation technologies and of the power market leads nowadays to complex, high-efficiency plants with combined cycles, co-generation, and so on. Modelling and simulation tools must allow to analyse problems that become more and more complex, with lower costs than those accepted in the past.

In recent years, object-oriented methodologies for dynamic modelling of physical systems have emerged, culminating in the definition of the Modelica language (Mattson et al. 1998), which is now supported by efficient and reliable software packages such as Dymola. Several applications of Modelica to thermo-hydraulic systems such as boilers and heat generating equipment exist in the literature, see e.g. (Tummescheit et al. 2000; Fritzson 2003). Modelica libraries proposed for this kind of systems, see e.g. (Tummescheit et al. 2000; Elmqvist et al. 2003; Steinmann 2003), are structured in different ways, depending on the modelling purpose they are meant for (e.g., component sizing, system studies, control commissioning, and so forth). Libraries also differ for the specific application domain(s) they are targeted to (e.g., power plants, HVAC systems, and so on).

Based on the work reported in (Casella and Levi 2003a,b), this paper addresses the adoption of the Modelica framework for the modelling and simulation of thermo-hydraulic power generation processes with enough fidelity for detailed system studies. The paper then presents the ThermoPower Modelica library, designed with the approach illustrated herein, and available at www.elet.polimi.it/upload/casella/thermopower.

DESIRED FEATURES AND LIBRARY PRINCIPLES
A modelling and simulation environment for thermohydraulic process should possess some particular features that, to the best of the authors’ knowledge, are not currently available all together in any environment. The first, mandatory feature is modularity, and the possibility of hierarchical composition of models. The level of detail should be scalable, as some parts of the process may need representing with different level of detail depending on the simulation purpose.

Partial differential equations should be supported in their native form, as many essential elements of thermo-hydraulic processes are described by distributed models.

Rich libraries of models, at least of the most common components, should be available. The modelling language should be transparent, i.e., as near as possible to the original equations, to avoid error-prone and hard-to-document elaboration.

The user should be allowed to easily incorporate his/her own know-how in the modelling library, by either modifying or creating component models, and the structure of the library models should allow the extension of the modelling hypotheses where needed.

Support should be provided for dynamic simulation initialisation, as this is one of the most critical tasks in dynamic simulation: note that, in the context addressed herein (contrary, for example, to the mechanical domain), models are very sensitive to erratic initial states, and it is not easy to define (not to mention to compute) a ‘rest state’. The literature is too often silent on the initialisation problem, while the authors believe that an effective solution of it is vital for a wider acceptance of dynamic simulation tools.

Libraries of functions to compute the physical properties of the substances involved in the processes should be available.

The modelling language should support Computer Aided Control System Design (CACSD), and be able to describe the control system with a different degree of detail depending on the purpose of the analysis: the formalism used to model the control system should be consistent with those used by CACSD tools, and the possibility of exchanging data with the field is highly desirable.

The environment should allow to easily compute and extract data which are relevant to the control system design, e.g. open loop step responses or linearised input/output dynamic equations.

Finally, besides being used for off-line simulation, a dynamic plant model could be used for real-time activities: these activities should be supported with the least possible effort by the user.

Coming to available environments, the modularity requirement rules out causal modelling approaches, in which
inputs and outputs are defined a priori for every model. Having established this, the modelling and simulation approaches of interest for thermo-hydraulic processes can be broadly divided in three categories.

Specialised simulation packages, such as APROS or Pro-TRAX, heavily exploit the particular structure of the equations describing thermo-hydraulic processes, employing ad-hoc algorithms for their numerical integration. The modelling paradigm is modular and large libraries are available, but there is no clear separation between the mathematical models of the components and the numerical integration algorithms, which impairs transparency.

Generic equation-oriented languages, such as gPROMS, provide primitives for the building of complex plant models starting from the single components, but there is no formal need or incentive towards model abstraction, encapsulation, and use of standard connectors, nor towards building reusable component libraries.

Generic object-oriented languages, like Modelica, strongly emphasise the inheritance mechanism and the encapsulation concept. Abstract, general “parent” models are first defined, and then more specialised “child” models are derived from them by adding equations which specify their behaviour completely. Component models are meant to be connected through rigorously defined interfaces (connectors). The development of reusable libraries of models is thus strongly supported.

The ThermoPower library was designed as an attempt to respond, as completely as possible, to the above requirements.

Its principles are illustrated in the following, motivating the adoption of Modelica as the host environment. A more detailed discussion is reported in [Casella and Leva 2003a], to which the interested reader is referred, while a longer explanation of the modelling principles adopted in the library can be found in [Leva and Maffezzoni 2003].

Use of first-principle equations. The equations of the library models are derived from mass, energy and momentum balances, and (when necessary) from well-established empirical correlations. All variables can be given a physical meaning.

Openness and transparency. The features of Modelica are exploited to obtain a code that tightly matches the way equations are written on the paper. This facilitates documenting and maintaining the library, and allows the users to understand exactly what they are simulating. Also, Modelica allows to investigate different modelling options quickly, and the inherently open nature of the environment permits modifications and improvements of models with a limited effort.

Readability-reusability trade-off. The inheritance mechanism is used sparingly, and with great care. Even though inheritance appears very attractive when structuring a component library, it is very difficult to define sufficiently general basic models in the application domain addressed here. Moreover, in a complex hierarchy of models, modifying the equations of some ancestor could have unexpected effects on the siblings, potentially impairing readability, not to say correctness. Since even fairly complex models can be described with a few dozen lines of code, it is advisable that the behaviour of a single component be described in a single place, rather than scattered through many different classes.

Partial Differential Equations. For the purposes of this work, models based on 1-dimensional partial differential equations are needed, which are not supported by Modelica in their native form. Therefore, such equations are reduced to ordinary differential ones by appropriate methods (e.g. finite volumes, finite elements) prior to their insertion in a Modelica model.

Standard interfaces. In the library, connectors are designed so as to be totally independent of the modelling assumptions adopted for the component. The same terminals are used no matter whether the fluid is assumed to be one- or two-phase, the model is lumped- or distributed-parameter, the momentum balance is static or dynamic, the cross-sectional fluid velocity distribution is uniform or not, the phases in two-phase flows are assumed to have the same velocity or not, and so forth.

Flexible level of detail. Encapsulation is exploited to allow for models with different degrees of detail, fully interchangeable with each other. This means that, in different situations, the same component or part of the plant can be modelled with different detail levels, with a small effort on the part of the analyst.

Substance property calculation. Medium models for water, steam, and ideal gas mixtures are already provided by the free Thermofluid library (Tummescheit 2000) and by the forthcoming Modelica Media library (Elmqvist et al. 2003). Simulation efficiency could possibly be increased by using third-party property calculation packages written in C or FORTRAN. The library is open to such extensions.

Models for different fluids. It would be possible to make the equations of a component highly independent of the fluid contained, thus reducing the total number of library components. However, in the thermo-hydraulic domain there are basically two cases, leading to different modelling choices. When changing the fluid only means changing the way its properties (temperature, density, etc.) are computed, while the equations describing the physical component remain the same, inheritance is a good choice. This is the case, for example, when a certain ideal gas mixture is replaced by a different one. When changing the fluid (also) means changing the structure of the component equations, instead, as it is often the case with water/steam models versus gas models, attempting to write them in a ‘general’ form leads to a significant complication. In such a case, the authors’ opinion is that it is preferable to write specialised models for the different fluids.

DEVELOPED MODELS

General Considerations

At the time of this writing, the library consists essentially of three sub-libraries, or packages: ThermoPower.Water, containing models with water or steam as working fluid, ThermoPower.Gas, containing models with ideal gases as working fluid, and ThermoPower.Thermal, containing models of heat transfer phenomena and components. Many models in the first one (e.g., sources, sinks, valves) have a counterpart with the same name in the second, which is based on the same general principles, even though the actual equations may be somehow different. In the following discussion, for the sake of brevity, the two counterparts will be described together.

While the models in ThermoPower.Water are bound to the water/steam medium, those in ThermoPower.Gas are parametric with respect to the gas model, which can be a
fixed composition ideal gas, or any mixture of ideal gases. The equations of the gas models are thus written in order to be applicable in both cases.

The models can be connected to each other by means of standard connectors. The FlangeA and FlangeB connectors, containing pressure, mass flowrate, enthalpy and possibly composition of the fluid, are used to connect the fluid inlets and outlets. The DHT distributed heat connectors, containing vectors of temperatures and heat fluxes, can be used to connect 1-dimensional objects exchanging thermal power, such as a fluid flow and a metal wall in a heat exchanger.

**Boundary Conditions**

Ideal pressure sources and sinks have been defined (SourceP, SinkP), as well as mass flowrate sources and sinks (SourceW, SinkW). Note that the difference between source and sink is purely conventional, as both of them can handle flow in either direction. Hydraulic and thermal variables, as well as the composition in the case of gases, can be either constant, or determined by external signals.

**Branching Components**

Flange terminals only support connection of two components; therefore, the FlowJoin and FlowSplit components are provided to model flow branching. The model are based on static mass and energy balances equations, supporting all the feasible flow directions and avoiding numerical singularities.

**Elementary Physical Components**

**Pressure drop**

The PressDrop model provides the model for a generic pressure drop proportional to the kinetic pressure. The equation is modified by adding a small linear term, to avoid singularities with small or zero flowrates, thus reading:

$$p_{in} - p_{out} = \frac{K_f(|w| + K_f)w}{\rho}$$  \(1\)

The same modification also applies to all the other models containing pressure drops.

The simpler PressDropLin model assumes a pressure drop which is proportional to the mass flowrate.

**Valves**

The WaterValveLiq, WaterValveVap and GasValve models are based on the standard IEC 535 sizing equations for valves with liquid water, steam and gas flow, respectively. Choked flow can be modelled in all cases, as well as check valve behaviour. Flow reversal is supported, avoiding numerical singularities for small or zero pressure drop. The opening characteristic can be customised.

A simplified valve model, where the pressure drop is proportional to the flowrate and to the valve opening, is provided by the ValveLin model.

**Mixers, collectors, tanks**

The Mixer and Collector models are based on standard mass and energy balances, assuming uniform pressure and temperature in the control volume; they differ only by the number of connecting flanges. Heat exchange with the metal wall can be also accounted for.

The Tank model describes an open tank, while the Accumulator models a gas-pressurised water accumulator with gas charge and discharge valves.

**Pumps**

Since storage of mass and energy are negligible, the PumpMech model is expressed by algebraic characteristic equations derived from the manufacturer’s design data, that relate the pump head and the mechanical power consumption to the rotation speed and to the volumetric flow rate. Besides the inlet and outlet thermo-hydraulic connecters, this model features a mechanical connector to be bound to the electromechanical model of the driving motor. The simpler model Pump lacks the mechanical part and takes the rotation speed from a signal input.

**Compressor, turbine**

The simple models Compressor and Turbine are based on the so-called first and second characteristic equations, which relate the mass flow rate and the isentropic efficiency to the pressure ratio, inlet density, inlet temperature, and shaft speed, by means of suitable non-dimensional groups (flow number and blade Mach number) and user-customisable characteristic functions. Besides the gas inlet and outlet flange connectors, a standard mechanical connector for the shaft is provided, which is compatible with the standard Modelica mechanical models. Due to the low volume/flowrate ratio, mass and energy storage can be neglected, leading to purely algebraic models. More detailed row-by-row dynamic models could be easily written if needed, e.g. to provide sophisticated modelling of surge and stall phenomena, provided all the relevant dimensional data are known.

**Combustion chamber**

The simple GasCombChamber model is based on lumped-parameter mass and energy dynamic balance equations, assuming complete combustion. The lower heating value of the fuel can be either computed from its composition, or be directly supplied by the user.

**Gas Turbine Unit**

Detailed model of single-shaft or multiple-shaft Gas Turbine Units can be obtained by connecting the compressor, turbine, and combustion chamber models with the standard Modelica models of the shaft and mechanical/electrical load models. This level of detail is required for an accurate dynamic modelling of the GTU, in particular when regenerative cycles are present.

For simpler modelling tasks (e.g., when the flow characteristics of the compressor and turbine are not known), the GTU model is provided, which is based on global unit characteristics rather than on the separate characteristics of the turbine and compressor.

**Drum**

The Drum model is the core of drum boilers models (Maffezzoni 1992). In order to describe correctly the dynamics of fast transients, the model does not assume that the liquid and vapour phase are in thermodynamic equilibrium, i.e., at saturation state. The model equations (Casella and Leva 2003b; Leva and Maffezzoni 2003) are based on separate mass and energy dynamic balance
equations for the liquid volume and the vapour volume, and on a dynamic energy balance for the metal wall. With reference to Figure 1, the two-phase flow from the risers \( w \) is separated at the drum inlet and goes part in the liquid volume and part in the vapour volume; the feedwater flow \( w_f \) enters the liquid volume, and the downcomer flow \( w_d \) consists of a mixture of the feedwater flow and of the liquid inside the drum; the steam flow \( w_s \) goes out of the vapour volume. Heat transfer takes place between liquid and vapour \( Q_{lv} \), metal wall and liquid \( Q_{wl} \), metal wall and vapour \( Q_{wm} \), metal wall and external environment \( Q_{we} \), proportional to the temperature differences. Heat and mass transfer takes place between the two phases through a bulk condensation flowrate \( w_c \), a surface condensation flowrate \( w_{cvs} \), and a bulk evaporation flowrate \( w_{ev} \).

**Figure 1: Steam Drum**

**Building Blocks for Complex Components**

1-dimensional fluid flow
The Flow1D model describes the 1-dimensional flow of single-phase water in a tube of constant cross-section. The basic equations are 1-dimensional distributed-parameter mass, momentum, and energy balances:

\[
A \frac{\partial \rho}{\partial t} + \frac{\partial (\rho w)}{\partial x} = 0
\]

\[
\frac{\partial w}{\partial t} + A \frac{\partial (\rho w)}{\partial x} + \rho g A \frac{dz}{dx} + C_f \frac{\omega}{2} \rho A^2 w |w| = 0
\]

\[
\frac{A h}{\partial t} + A u \frac{\partial h}{\partial x} - A \frac{\partial \rho}{\partial t} = \omega \varphi
\]

where \( \rho \) is the fluid density, \( w \) is the mass flowrate, \( p \) is the pressure, \( A \) is the tube cross-section, \( g \) is the acceleration of gravity, \( z \) is the elevation, \( C_f \) is the Fanning friction factor, \( \omega \) is the tube perimeter, \( u \) is the fluid velocity, \( h \) is the fluid enthalpy and \( \varphi \) is the heat flux entering the tube across the lateral surface. These equations describe the fast dynamics of pressures and flowrates, together with the slower dynamics of the enthalpies: they are then discretised with the finite element method, assuming uniform pressure and flowrate throughout the tube length, and describing the distribution of the enthalpy with \( N \) equally spaced nodal values (for more details, see Casella and Leva 2003a). Among the relevant features of this model, the following ones are worth mentioning: flow reversal is fully supported; the dynamic momentum \( \omega \partial \dot{\varphi} \) can be switched off to avoid fast pressure oscillations caused by the wave dynamics; the \( C_f \) coefficient can be either constant or computed by Colebrook's equation; the compressibility effect resulting from the finite volume approximation of (2) can be associated to either the upstream or downstream pressure; a bank of identical tubes in parallel can also be modelled.

The Flow1D2ph model can also deal accurately with two-phase flow; though it is based on the same equations (2)-(4), the significant differences with respect to modelname {Flow1D} suggest writing two completely independent models.

The Flow1DDB model extends Flow1D through inheritance, by adding the computation of the heat transfer coefficient \( \gamma \) by Dittus-Boelter's correlation; correspondingly, the DHT connector (which is replaceable) is substituted by the extended DHTrte connector, which makes the values of \( \gamma \) visible to the outside.

When dealing with two-phase flows, Flow1D2phDB and Flow1D2phChen can be used. In both cases it is assumed that the critical heat flux corresponds to a given steam quality. If the fluid state is liquid, or steam having a higher quality than the critical one, the heat transfer coefficient is computed by Dittus-Boelter's correlation; otherwise, the boiling heat transfer coefficient is assumed as a given constant for the former, or computed by Chen's correlation for the latter.

Finally, finite element versions of these models (Flow1D2phChen, Flow1D2phChen2ph) have been successfully developed; see (Casella and Schiavo 2003) for details.

**Metal wall**
The MetalWall model describe a generic cylindrical metal wall, accounting for the thermal resistance due to heat conduction and for the heat storage due to thermal capacity, the latter being lumped in the middle of the wall thickness. More sophisticated models could be derived to better reproduce the actual radial temperature dynamics, e.g., in thermal stress studies.

**Heat exchange modules**
The heat flux exchanged between two (or more) objects, such as a fluid flow and a metal wall, is in general a function of the corresponding surface temperatures; therefore, it can be computed by a model interfaced via modelname{DHT} connectors. The ConvHTe and ConvHTc models provide simple examples for co-current and counter-current 1D configurations, with given heat exchange coefficient \( \gamma \). ConvHTe_htc extends the former by using a variable value of \( \gamma \), provided by the connected object through its DHT_gamma connector. More complex configurations can be easily described with a few lines of code.

**Complex Physical Components**
A whole range of heat exchanger models can be assembled using the components described in the previous section, depending on physical configuration, operating conditions and desired degree of detail. None of these models probably deserves to be included in the library as such; if a specific aggregate model is to be used many time, the user can easily define it as a new model inside his plant model. Some of them may nevertheless be included in the library to serve as examples.

**EXPERIMENTAL VALIDATION**
All the library models have been tested in very simple configurations, to check their correctness. Besides that, the
ThermoPower library was validated against experimental data coming from the physical model of the evaporating section of a heat-recovery boiler, with a power scaling factor of 1:600.

Details on this validation are given in (Casella and Leva 2003b). Suffice here to say that the simulation model represents the circulation loop in detail, while non-equilibrium phenomena are accounted for in the model of the drum. The model has 50 state variables and 288 nontrivial equations, showing that the proposed approach allows to treat efficiently cases of realistic complexity.

As an example, a sample of the validation tests is reported here, namely a negative throttling valve step leading to a 13% pressure reduction at low load. Figures 2 and 3 show the drum pressure and level transients: note the good agreement between the model output and data.

In particular, the non-equilibrium phenomena represented in the drum model allow to reproduce both low- and mid-frequency dynamics in the pressure responses correctly. Also the effects of thermal exchanges between the fluid in the drum and the drum metal wall were investigated, showing that the corresponding heat transfer coefficient has a significant influence on the superheated steam temperature. This phenomenon is often neglected in the simulation models proposed in the literature.

CONCLUSIONS AND WORK IN PROGRESS

After discussing the features of a modern dynamic simulation environment for thermo-hydraulic power generation processes, general guidelines of an object-oriented modelling approach have been proposed, and an open Modelica library adopting that approach has been presented.

The ThermoPower library emphasises model readability and extensibility; it contains a limited number of components, and nevertheless allow modelling a wide range of different physical components. The object-oriented approach, and in particular the Modelica language, used as described herein, allowed translating sophisticated modelling concepts into working code with remarkable ease.

The library and the briefly mentioned benchmark boiler model, together with the experimental data set, have been put in the public domain under the Modelica license, and are available on the library web page. The library is currently been applied for the modelling of fossil-fired plants (conventional, combined-cycle and combined heat & power) as well as nuclear plants.

REFERENCES


http://sourceforge.net/projects/thermofluid
PROGRAMMED TRAINING SCENARIOS IN POWER PLANTS’ SIMULATORS

Endre VÉGH, Laura BÜRGER
KFKI Atomic Energy Research Institute
1525 Budapest 114 P.O.Box 49, HUNGARY
E-mails: vegh@sunserv.kfki.hu, buerger@sunserv.kfki.hu

KEYWORDS
Computer systems, Intelligent simulation environment, Simulator upgrade, Simulation interfaces, Training

ABSTRACT
This paper describes an automated training session control containing all malfunction initializations, parameter modifications and external operators’ actions. These simulation control actions are stored in scenario scripts. With the scenario scripts the execution of the simulation is modified. In every simulation cycle the scenario scripts are evaluated before the execution of the model programs. Each event in the scenario can be prompt, delayed or conditional depending on the fulfillment of a logical expression. The paper describes the structure of scenario scripts in the Paks Full-scope Training Simulator, moreover, the construction and the execution of the scenario scripts are also presented.

INTRODUCTION
In nuclear power plant simulators periodical retraining of the operating personnel of the plant is of primary importance. During a training session, the operators are faced to unknown situations to be solved and their reactions are evaluated. Since the instructor has to concentrate on the trainees’ performance, therefore he/she must be relieved from the control of the session. Because a training session contains a series of events, the control of session yields automatic execution of these events. In this paper we shall call as “training scenario” all of the events, to which the operators’ are faced during a training session, thus session control means automated execution of a predetermined training scenario.

The full-scope training simulator of the Paks Nuclear Power Plant (Hungary) has been operating since 1988. Because the training requirements are permanently growing and the technical possibilities are constantly improving, this simulator has been upgraded many times in the last ten years. First its model system was widened in 1994 to contain sever accidents of the primary circuit (Végh at al 1994; Végh, Jánosy and Hózer 1995), then the original simulation computer was replaced by a DEC AlphaServer 2000 in 1997, and at last the whole Control Room Interface System was upgraded in 2002 (Végh at al. 2003). Until now only the instructor’s system remained intact and its reconstruction is in progress. Last year the Technical Design of the new instructor’s system was completed, and the programmed training scenario system belongs to this new system.

SCENARIO COMPONENTS
Each training scenario can be divided into single events. An event is a single modification in the functions describing the operation of the simulated technology. Thus an event happens when e.g. a parameter starts to change according to a given time function. However, during its change, each time step will not be regarded as a new event, because only the parameter’s value and not its behavior changes.

In general, an event consists of a command and a condition part. The command determines what will happen, the condition describes when.

COMMANDS
Commands can be malfunctions, instructor’s actions and parameter modifications. A malfunction describes a failure in the technology: a break, a valve stuck, an erroneous sensor reading, etc. Instructor’s actions are needed, when a controller has to be operated, which is inaccessible from the control room of the simulator. Parameter modification describes the changes in the operating conditions of the plant e.g. the change of the plant load or the warming of the cooling water. In most cases, these parameters are external parameters of the simulator. Malfunction commands are executed in the modeling system, while the other two commands need database handling operations.

CONDITIONS
An event can be conditional or unconditional. An unconditional event can be prompt (i.e. the event is executed immediately at the start of the scenario), or delayed (i.e. the event is executed when a given time
interval from the scenario start passes). A conditional event occurs when a logical relationship is fulfilled e.g., a pipe breaks when a pressure is greater than a given value. In the following, the logical condition of an event will be called as the trigger function of the given event. The trigger functions are stored in the trigger library and they are referred by their names.

**SCENARIO DESCRIPTION**

In the following part, we shall describe the training scenario system of the Paks Full-scope Training Simulator. In our solution, the training scenarios are easily readable scripts. It is an important feature, because in this way the scenarios are self-documented. Every event is a line in the scenario script. The different scenario scripts are stored in scenario files. The number of the scenario files is not limited. When the instructor loads a new initial condition file, he/she can load a scenario file, as well. After having loaded a scenario file, an interpreter program stores the loaded scenario script into its executable representation (see later) in the memory. In our system, the general form of an event is the following:

**Delay={value} | Trigger={name} -> Command**

where “value” is a number, and “name” represents a valid file name in the trigger function library. The meaning of this script is the following: the command is to be executed when the value of the trigger function is true or when the time delay from the scenario start passes. Both operative words can be omitted. If “Delay” is missing, the event is conditional. If “Trigger” is missing, the event is delayed. If both operative words are missing, the event is prompt, i.e. it is executed immediately at the start of the scenario.

**SCENARIO COMMANDS IN THE PAKS SIMULATOR**

The general form of a command in our system is the following:

**KeyNumber {Variable} {Value Transition}**

Where

- “Key” is a keyword, representing the command type
- “Number” is an identification code, specifying the actual command
- “Variable” is a name of an entity in the database
- “Value” represents the desired value of the addressed variable

- “Transition” describes the way, how the desired value is to be reached (as jump or ramp function)

In our system the following keywords are used:

- MF = malfunction,
- LOA = instructor’s action,
- EXTP = external parameter modification.

Parameters in {} brackets can be omitted, because e.g. the specification of the “Loss of load” malfunction does not need any variable.

**TRIGGER FUNCTIONS IN THE PAKS SIMULATOR**

Trigger functions consists of operands and operators. Operands are variables in the database of the simulator and are referred by their unique names. Operators are summarized in the following table, where p1, p2, etc are operands, and i stands for the number of the operands (2÷6):

Table 1: Operators of the trigger functions

<table>
<thead>
<tr>
<th>Type</th>
<th>Operator</th>
<th>Form in the script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical</td>
<td>AND, OR</td>
<td>#AND(p1,p2,p3)</td>
</tr>
<tr>
<td>Relational</td>
<td>&lt;,&gt;,&lt;,&gt;=</td>
<td>(p1 &lt; p2)</td>
</tr>
<tr>
<td>Arithmetical</td>
<td>+,-,*,/</td>
<td>(p1 + p2)</td>
</tr>
<tr>
<td>Bitwise</td>
<td>if (p1 = = value)</td>
<td>%SZBE p1</td>
</tr>
</tbody>
</table>

From the logical operators only logical AND and OR functions are realized. In both cases, the length of the arguments is limited to six. Each element of the argument can be either an expression, or a logical variable.

Relational operators compare the values of two quantities. Only “less than” and “greater than” type relations are realized. Both quantities can be an arithmetical expression, a real variable, or a numeric constant.

Arithmetical operators represent only the four elementary operators connecting two quantities to each other. Both quantities can be an arithmetical expression, a real variable, or a real constant.

Bitwise operators form a condition on the state of control elements. They check the bit pattern of the state word of the addressed control element and produce logical true if e.g. the addressed valve is open. The arguments of these operators are integer status words of valves, motors and switches. The realized bitwise operators are listed in the following table.
Table 2: Bitwise operators in the Paks Simulator

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>%SZBE</td>
<td>True if the valve is closed</td>
</tr>
<tr>
<td>%SZKI</td>
<td>True if the valve is open</td>
</tr>
<tr>
<td>%SZKZ</td>
<td>True if the valve is in middle position</td>
</tr>
<tr>
<td>%MTKI</td>
<td>True if the motor does not operate</td>
</tr>
<tr>
<td>%MTBE</td>
<td>True if the motor operates</td>
</tr>
<tr>
<td>%KPBE</td>
<td>True if the switch is closed</td>
</tr>
<tr>
<td>%KPKI</td>
<td>True if the switch is open</td>
</tr>
</tbody>
</table>

Trigger functions contain information on their graphic representation, as well. The position of any component in the graphic representation of a trigger function has two coordinates (x,y). These coordinates are also stored in the trigger function scripts in { } brackets separated by the colon symbol. An example for a trigger function script is the following:

TR =
#ORZ{560:100} (#AND2{460:50} ((X0TD60S201N)
{320:35}, %SZKI X0TH30S202{460:150}),
(X0TX0L101{360:125} > 808.6{360:175}))

Trigger functions are always presented to the instructor in graphic form as directed graphs. In this graph, each component has an icon and lines connect the related components to each other. The colors of the lines represent the types of the components (blue means real, red stands for logical). Figure 1. presents all trigger function icons used in the Paks Simulator.

TRIGGER FUNCTION CONSTRUCTION

The construction of the trigger functions is fully graphical. The instructor selects a trigger function icon by clicking on it with the mouse, and it appears in the construction field of the display screen. In this field the icons can be dragged by the mouse to the desired position. This position is stored in the trigger function script in the x,y coordinates. By clicking on the output of an icon, it can be connected to the input of an other one with the mouse, if the input/output types are identical.

SCENARIO CONSTRUCTION

The scenario construction is graphical. Scenarios may be constructed when their execution is forbidden. The construction of a malfunction event is presented - as an example - in Figure 2. (See on the next page). At the upper left side one can select the desired malfunction group in a list; in this example “Primary circuit” is selected. When the group selection is done, all malfunctions of the primary circuit appear in the lower left list. When the instructor selects a malfunction (in this example it is: “Wrong position indication of a control rod”), the permitted addresses (control rod names) are presented at the right side. The instructor selects an object from this list, then he/she sets the delay, the desired parameter values by vertical scroll bars, and the name of the trigger function with a ComboBox. The constructed malfunction event is stored in the scenario script when the instructor clicks on the ▼ symbol above the scenario list.

EXECUTION OF THE MALFUNCTION COMMANDS

Malfunction scenario commands are evaluated in every simulation cycle before the execution of the model programs. At a given moment the defined malfunctions are stored in the Malfunction Table. This table contains status bytes describing the actual states of the defined malfunctions. The possible states of a malfunction are the following:

- Waiting, i.e. either the trigger function is false, or the time delay has not passed,
- Active, i.e. the starting condition has been met, but the execution of the malfunction has not yet started,
- Effective, i.e. the model program has started the execution of the given malfunction,
- Modified parameter, when the value of the malfunction parameter has been changed,
- Request for inactivation,
- Inactive, when the malfunction has been finished.
During the evaluation of the malfunction commands, the status byte for each defined malfunction is determined.

The malfunction programs are parts of the technological model programs. Every model calls a special function with the identification code of that malfunction which belongs to the given model. This function gives back the names of those objects, whose status byte is active, or its parameter is changed, or its inactivation is requested.

**EVALUATION OF THE TRIGGER FUNCTIONS**

In the run-time representation, every trigger function is represented as a binary tree. Every logical gate can be represented as a series of logical gates with two inputs. In this way, an operator has always two operands. The result is a binary tree, in which each operator and operand is regarded as a node of the tree. In the memory the binary tree is stored in a list structure.
Trigger functions are evaluated each time, when scenario commands are executed. The evaluation starts at the top node (TR1). At first the left pointer branch is evaluated. The evaluation is a recursive algorithm. Recursion is needed until the root node, where the operands of the operator are either single variables or constants. The result is always a logical value, which is stored in the Trigger Function Table. The values of this table control the activation of the conditional scenario commands.

EXECUTION OF THE DATABASE RELATED COMMANDS

Database related commands are instructor’s actions and parameter modifications. Instructor action commands generate fictive control room input events identical to the real ones. The automatic models cannot distinguish these events from the real ones generated by the control room communication subsystem of the simulator, thus the models execute the instructor’s action commands exactly the same way as the control room inputs.

Parameter modifications are executed in every simulation cycle before the execution of the model programs. The actual value of every modified variable is calculated by that task which evaluates the status bytes of the malfunction commands.

CONCLUSION

The described automatic scenario execution system is at present under construction. The system will be ready by the end of this year and it will be installed at the Paks Full-scope Training Simulator in January next year. After a two-week long testing period, the automatic scenario execution system will be used in the plant operators’ training.

ACKNOWLEDGEMENT

The authors wish to express their special thank to Mr. László Dereze, deputy head of the Simulator Centre of the Paks Nuclear Power Plant and to the instructors of the simulator for their collaboration on various aspects of this work.

REFERENCES


AUTHORS’ BIOGRAPHY

LAURA BÜRGER received her M.Sc.E.E. degree at the Technical University of Budapest in 1958. She is member of the European Nuclear Society. Her research field is the application of computer technique in the nuclear industry. Previously she was interested in computerized methods of disturbance analysis. She took part in the Jánossy price awarded team realizing a computer based closed loop reactor control system on the Budapest Research Reactor. Presently her main interests are: real time data collecting and processing systems, simulators, etc.

ENDRE VÉGH received his M.Sc.E.E. degree at the Technical University of Budapest in 1961. At that time he joined the Central Research Institute for Physics (KFKI), where he was engaged in the construction of nuclear instruments and in the development of different computer systems for the nuclear industry (reactor information systems, power plant simulators). In the period of 1980-1995 he was the head of the Simulation and Reactor Control Department. He is a member of the Hungarian Nuclear Society. His research fields include reactor core surveillance, plant computers, simulation of nuclear power plants.
An Experience Report: RT-CORBA in Nuclear Power Plant Simulators

Manuel Díaz
Daniel Garrido
University of Málaga
Department of Languages and Computing Science
CP.29071 Málaga, Spain
E-mail: {mdr,dgarrido}@lcc.uma.es

KEYWORDS
Communications, Nuclear engineering, Training, Interactive simulations, Real-time simulation.

ABSTRACT
The application of new technologies and programming tools represents a challenge and an economic risk for companies, which not all are prepared to assume. The Real-time CORBA specification is a new technology that allows the development of distributed applications with real-time constraints supporting end-to-end predictability in fixed-priority applications where the management of the processor, communications and memory resources is possible. This paper presents RT-CORBA in the context of Distributed Nuclear Power Plant Simulators used for the training of future operators in a safe way. The Simulators are based on a complex architecture of simulation models with real-time constraints involving many different applications that allow the Full Scope Simulation of the Control Room of a Spanish Nuclear Power Plant.

INTRODUCTION
The development of simulators for Nuclear Power Plants is traditionally based on old techniques and languages like FORTRAN, Ada or C. However, new techniques and methodologies like CORBA (Henning and Vinoski 1999) could be applied in the development of these systems. CORBA is a middleware that allows the communication between objects in different hosts of the network in a transparent way where there is no difference between local or remote objects and where the differences of operating system, programming language, network technology, etc. are hidden.

This paper presents the extension of CORBA for real-time systems commonly known as RT-CORBA (Schmidt and Kuhns 2000) into complex distributed simulators for Nuclear Power Plants with real-time constraints. The paper is specifically situated within the project of simulators for a Nuclear Power Plant sited in Trillo (Guadalajara, Spain) that during two years has allowed the carrying out of a Full Scope Simulator of the Control Room of this Plant.

The simulator is an exact replica of the Control Room of the Power Plant and its main intention is the training of future operators. The simulator takes care of all details, from physical artifacts like furniture, control panels, etc. to software, simulating the applications running in the Power Plant. The kernel of the simulator includes simulation models with real-time constraints, which proportionate the values of the distinct signals and variables needed by the rest of hardware and software components.

The software described in this paper has been carried out in a shared project between the company Tectomat S.A. and the department of Languages and Computing Science at the University of Málaga. The work has been mainly related to the building of new software with special emphasis on the communication’s infrastructure based on RT-CORBA and in the building of software components (Szyoperski 1999) that could be reused in future projects.

The rest of the paper is organized as follows: The hardware architecture is presented in the following section. Next, the software architecture and finally the RT-CORBA issues with some conclusions.

SYSTEM ARCHITECTURE
The Trillo Simulator Project is divided into two simulators that determinate the hardware architecture and the physical infrastructures. The first simulator is the denominated Interactive Graphic Simulator (SGI), which through graphic applications (see Figure 1) allows the training of future operators. The second simulator is the denominated Full Scope Simulator (SAT), exact replica of the Control Room of the Power Plant (see Figure 2). These simulators share parts of the hardware architecture and a large part of...
the software components, specially the components related to communications.

![Figure 1. Example of SGI application](image)

The high-level hardware components of SAT and SGI simulators include: Simulation Computers, Plant Process Computer, Instructor Console and Physical panels.

The Simulation Computers are responsible for the simulation process executing the simulation models and providing data to the rest of software and hardware components. The Plant Process Computer is a complex subsystem responsible for carrying out the simulation of this real component of the Control Room and whose main task is the monitoring of data values and presentation of alarms, graphs and reports. The Instructor Console only exists in the context of the simulators, and it allows the creation of simulation conditions that have to be solved by the students. Physical panels are exact replicas of the existing in the Control Room. They are situated in a large room (17x18 m.). Operators of the Power Plant carry out their actions mainly through these panels, which have a lot of indicators, hardware keyboards, valves, etc. A large wiring is required for all the simulation process, allowing the connection between hardware and software components.

The SGI simulator additionally includes the needed hardware for the student posts of the simulator. Basically a student post allows the practice of all the areas of simulation in a comfortable way with several monitors for each post and graphical applications.

A scheme of the hardware equipment of the simulator and the physical installations can be seen in Figure 3. The Instructor room is at the top of the figure. In this room the instructor will manipulate the simulation session modifying the state of the simulation. Next to this room is found the Computers room where some of the previously described computers are situated. Finally, depending on the used simulator (SGI or SAT) the student posts will be found into the SGI room and there is an annexed room with the Physical panels for the SAT simulator.

![Figure 3. SGI-SAT Simulator](image)

**SOFTWARE ARCHITECTURE**

The developed software for the SGI-SAT project is divided into three well differentiated phases. First of all, the adaptation and documentation of previous software was performed for its use with CORBA, UML and object-oriented methodologies. Secondly, the development of the new applications needed for this concrete project. The third phase consisted of the interconnection of the updated software with the new applications, allowing, for example,
that the PPC could retrieve simulation data updated by the simulator kernel.
Two different development environments were mainly used: Unix with IRIX 6.5 of Silicon Graphics and Windows with Microsoft Visual C++ 6.0. Different languages had to be used: C++, FORTRAN, Java, etc. In the case of CORBA
the chosen implementation was TAO 1.2. (Levine et al. 1998), freely available CORBA ORB that due to
characteristics like predictable timing and robustness is very adequate for its usage in real-time applications.

The main application and tools that form the simulation environment of SGI and SAT are described in the following
subsections.

**Simulator Kernel**

SETRU is the kernel of the different simulators of Tecnatom S.A. providing an execution environment for the simulation
models.

The simulation models are responsible for the precise simulation of physical components of the real system, like
valves, sensors, actuators, etc. providing a set of simulation variables which will represent the physical components and
which can be queried, modified, etc. with the rest of applications.

The models use RT-CORBA and they are distributed between different hosts on the network. The priority of a
simulation model is determined by the number of executions in a second. This number determinates the periodicity of the model and its priority.

**Debugging and Supervision Tools**

The simulation kernel is supported by several applications. Two very common tools are DESI (Variables Debugger) and
SPV (Supervisor). DESI allows the query and modification of variables associated with the simulation
models, being very useful for the validation of these models. On the other hand, SPV allows the modifying of
different simulation aspects like executed models, timing control of the models, etc.

**SGI Displayers**

The SGI Displayers allow the training of operators in a classroom differing the simulation in the SAT simulator,
where the different components of the Control Room are directly manipulated. A SGI Displayer allows the visualization of graphical sheets with the different
components existing in the Control Room. An example of SGI application is shown in Figure 1.

**SGI Displayers** are organized into an instructor/student scheme, where the instructor can carry out the same actions
that students can, and additionally other actions related to the management of the simulation session.

**PPC**

The goal of this subsystem is to report the state of the plant in the Control Room through alarms and graphics. It
consists of three computers: Supervision Computer (responsible for the supervision of new alarms), Core
Protection System Warning Computer (responsible for the supervision of own alarms of the nuclear reactor) and
Criterion computer (allows the safe execution of periodic tests over physical components).

**Javi**

New application, which has reused previous components specially related to communications, using the same
interfaces as the rest of components. Javi allows the graphical representation of the state of the plant through
color codes and has been developed using Java-2D and Java-3D. Figure 4 shows the state of a Plant through a Javi
sheet.

![Figure 4. Javi application](image)

**COMMUNICATIONS ISSUES**

The SGI-SAT project is composed by several soft real-time applications, which simulate the behavior of the Control
Room of the Power Plant. CORBA is the basis for the communications and specifically RT-CORBA, the
extension of CORBA for real-time programming. Logically, RT-CORBA cannot convert non-real-time operating
systems in real-time operating systems but in our system, Irix or Windows completely fulfill the real-time constrains of the system. In the following sections the main contributions using RT-CORBA in the field of distributed simulators will be described.

Simcorba/Receiver

Simcorba is responsible for the transmission of simulation data to the rest of client applications. It is closely related to the Receiver component; responsible for receiving the data and letting them be accessible for the application using the Receiver.

Receiver and Simcorba use the CLIENT PROPAGATED model of RT-CORBA where the requests of the clients are performed with the same priority (CORBA priority) in all the system. The data sent by Simcorba are closely related to the “post type” that was indicated in the registry of the Receiver. The post type is determined by the application that uses the Receiver component, existing types like PPC, SGI, CDI, etc. Furthermore, each type has its own priority. Figure 5 shows an example of Simcorba/Receiver with SGI and CDI clients where the priority of the CDI client is the highest.

![Figure 5. Priorities and post types](image)

Figure 6 shows a test relative to the response time of Simcorba. The test has been performed on Irix 6.5 with a high priority client (32,000) and a low priority client (10) with a different number of additional low priority clients, which are successively started.

The graph shows the evolution of the response time of both clients when the number of additional clients is growing. So, the tests starts only with one client (CORBA priority 10 or 32000) and subsequently the number of additional low priority clients is raised. The response time of the high priority client is maintained in opposition to the low priority client whose response time raises in a lineal form.

Simulator Kernel

SETRU and the simulation models make up the simulator kernel. The simulation models are executed on different nodes of the network and their results are updated in the simulator kernel allowing to Simcorba and other tools to recovery updated data. Very common models are TRAC (thermo hydraulic model) or NEMO (neutronic model). Each model has different real-time constraints and so a different priority. The communication between SETRU and the tools use a client/server model where the tools (DESI, SPV) send commands and the models update the variables.

Thread pools of RT-CORBA are used to guarantee resources and the priority of the models is maintained using the CLIENT_PROPAGATED policy of RT-CORBA. Furthermore, the synchronization in SETRU is a fundamental aspect, so there are mutual exclusion zones in different parts of the kernel used by the models and the tools that have been protected through RT-CORBA mutexes.

Figure 7 shows an example of SETRU with several models and tools connected and the CORBA priority of each model and tool.

![Figure 7. SETRU with models and tools](image)
SGI Displayers

There are two types of SGI Displayers: instructor post and student post. The instructor is responsible for managing the simulation session; activating the student’s posts in an automatic manner, manipulating the simulation, etc. Each SGI post uses a Receiver component for the communication with the simulator.

The instructor configures the student’s posts. After configuring the posts, the Implementation Repository of TAO automatically launches the SGI Displayers of the students. The protocols between instructor and students require that all the SGI Displayers act like a CORBA server. These servers are activated with the SERVER DECLARED policy where the invocations are made at the priority indicated in the server. The connections are pre-established guaranteeing the needed resources. Finally, RT-CORBA timeouts are also used so the instructor cannot be blocked by bad connections with the students or dead posts.

PPC

The PPC subsystem has several Physical panels, computers controlling the state of the Plant, screens (at least 10) and keyboards associated with these screens. Figure 8 shows the architectural design of this subsystem.

The computers periodically (at least 4 times in a second) check for the existence of new alarms and provide these data to the PPC Kernel. These alarms have been provided by the Receiver component from Simcorba. The execution cycle of the PPC Kernel is responsible for processing these alarms and it creates the information in a suitable format for the screens. The screens recovery these data from the output manager and the data are shown.

The actions of the users are aperiodic and are processed by the keyboard manager contained into the PPC Kernel.

The mutexes of RT-CORBA are used again. In this case they are used among others reasons, to avoiding error accessing to lists of alarms. Finally, explicit binding with private connections is used to improve the connection between keyboards, screens and computers whose number is fixed and known in starting time.

CONCLUSIONS

The adoption of new technologies in a market dominated by costs requirements, development time, etc. is a big problem for the companies that want to adopt stable and secure technologies. In this paper, the adoption of new technologies and methodologies like RT-CORBA and UML has been presented in a project for the carrying out of Nuclear Power Plant Simulators.

The main contribution of the paper is the utilization of RT-CORBA in Distributed Simulators with real-time constraints. In this way, the development of the communication infrastructure is easier than with traditional techniques like RPC or sockets. The simulators are currently running and there are new projects where the developed software is being reused. For instance, Javi is a graphical Displayer of the Power Plant core that uses a great number of the components developed in the present project, specially the components developed with CORBA.

REFERENCES


ACKNOWLEDGEMENTS

We would like to acknowledge the help from the people of the División de Simulación at Tecnatom S.A. who contributed to the creation of this paper.
STEAM GENERATOR SIMULATION
SIMULATION OF THE STEAM TURBINE IN MATLAB ENVIRONMENT

Milan Cepák
Department of Control Engineering
Czech Technical University in Prague
Karlovo náměstí 13
121 35, Prague 2, Czech Republic
E-mail: cepakm@control.felk.cvut.cz

KEYWORDS
Energy, System identification, Model design, Continuous simulation.

ABSTRACT
The steam turbines are the basic units of power-producing thermal equipments. They represent the components where the general thermic processes take place. These processes consist in the transformation of the steam thermal energy to the prime mover mechanical power at high temperature and pressure. Their development is bound up with the amount of the energetic block losses. The suitable control algorithm can obtain considerable savings. But it should not be used without testing procedures because its possible failure might have the critical consequences. Thereby, it is important to simulate each control design on the system model. This paper describes the development of the mathematical model of the steam turbine in MATLAB. The next purpose is the testing of the designed model in the SIMULINK tool. The comparison of the data provided by the model with the real data represents the last goal.

INTRODUCTION
Recently, the thermal power plants contribute to the total production of the electrical energy by the substantial part. Thereby, serious attention is paid to the research and to the development of the steam turbines. Mainly the high power units are designed where the small increase of the efficiency is economic. Expect the amount, the price of the produced energy represents the other important aspect. In the literature, there are many algorithms describing the cost minimization problem. The method of the optimal load allocation (Šomvarský and Havlena 2002) used in the power engineering can be simply applied to the multipart steam turbines. Its principle is based on the price curves optimizations for every stage. This approach leads to the hierarchical control of the entire cascade. The subsystems coordination consists in the optimal settings of the throttling valve position of the appropriate part. This way, the required output with the minimal costs is assumed. Though, the control algorithm should not be used without the relevant testing because its possible failure might have undesirable effects. Each control design has to be simulated on the system model.

MATHEMATICAL DESCRIPTION
The modeled system is decomposed into two subsystems; the throttling valve, whose position is changed according to the power demand, and the turbine part, where the energy transformation of the heat content to the mechanical power takes place. Every subsystem is depicted by a set of physical equations (Ambrož 1980).

\[ \Delta p = k(y) \frac{F_{IT}^2}{2 \rho (p_{IT}, T_{IT})}, \]

where \( \Delta p \) [MPa] represents the pressure decrease, \( F_{IT} \) [kg/s] the steam flow, \( \rho \) [kg/m³] the steam density, \( p_{IT} \) [MPa] and \( T_{IT} \) [K] the input pressure and temperature and \( k(y) \) [-] is the coefficient parameterizing the pressure diminution by the valve position.

Figures 1: Scheme of the Two-part Backpressure Turbine

Throttling valve
The throttling valve serves for the control of the steam flow in the turbine part. There is a de-pressurizing of the input pressure to the adequate value. The throttling is often described as an isenthalpic process (Casas-Vasquez et al. 2001). But this approximation does not hold too much accuracy. The certain enthalpy loss is considered for more realistic throttling model. This loss changes according to the current steam flow. The intuitive principle admitting higher enthalpy losses for lower flows and vice versa may be accepted. Every valve keeps its own characteristic of the enthalpy loss \( \Delta h \parametrized by the steam flow \( F \).

The pressure decrease created by the throttling is given as
Turbine part

The conversion of the heat content to the prime mover mechanical power takes place in the turbine part. This thermodynamic cycle has its efficiency like other energy transformation processes (Salisbury 1974). The efficacy consists in many factors (Bloch 1995), e.g. used manufacturing technology influences the amount of the heat lost by the transmission through the part envelope. Losses developments depend not only on the structural properties but also on the current steam flow. The flow can be figured out via the pressure difference \((p_{IJ} - p_{in})\) at the part by famous Stodola equation (Brown 1997)

\[
F_{IT} = c \frac{P_h (1 + f)}{\sqrt{T}_{in}} \left[ 1 - \frac{P_{in}}{P_h} \right]^{3/2}, \quad (2)
\]

where \(F_{IT} \text{ [kg/s]}\) constitutes the steam flow, \(p_{IJ} \text{ [MPa]}\) and \(T_{in} \text{ [K]}\) the pressure and the temperature beyond the throttling valve, \(P_{in} \text{ [MPa]}\) the backpressure (the output pressure), \(c \text{ [-]}\) the unknown parameter and \(f\) is given by

\[
f = \eta \cdot \kappa, \quad \kappa = \frac{c_p}{c_v},
\]

where \(\eta \text{ [-]}\) represents the efficiency of thermodynamic cycle, \(c_p \text{ [kJ/kg.K]}\) and \(c_v \text{ [kJ/kg.K]}\) the heat capacity at constant pressure and constant volume. The variable \(f\) is often assumed to be constant \((f = 1)\); then, the equation (2) becomes simpler without the loss of generality. Known current steam flow permits to calculate the energetic losses. The amount of the steam, whose energy does not change to the mechanical power, is given by the equation of the lost work

\[
R_{div} = (h_{in} - h_{in}^*) F_{IT}, \quad (3)
\]

where \(R_{div} \text{ [kW]}\) corresponds to the lost work, \(F_{IT} \text{ [kg/s]}\) to the steam flow, \(h_{in} \text{ [kJ/kg]}\) to the part output enthalpy at the real \(\eta\) and \(h_{in}^* \text{ [kJ/kg]}\) to the enthalpy at the same place at \(\eta = 1\). The enthalpy \(h_{in}^*\) can be found by considering the thermodynamic cycle for the isenthalpic process.

The steam mass not participant in the energy conversion represents the second kind of the losses. The work of the lost heat passing through the turbine envelope is described by the entropy balance theorem

\[
Q = (T_{in} F_{IT} (s_{in} - s_{in}^*)) - R_{div}, \quad (4)
\]

where \(Q \text{ [kW]}\) is the work of the passing heat, \(s_{in} \text{ [kJ/kg.K]}\) the entropy beyond the throttling valve, \(T_{in} \text{ [K]}\) and \(s_{in}^* \text{ [kJ/kg.K]}\) the temperature and the entropy at the backpressure, \(F_{IT} \text{ [kg/s]}\) the steam flow and \(R_{div} \text{ [kW]}\) the lost work.

The primer mover power of the corresponding steam turbine part is given by the enthalpy balance formula

\[
J = F_{IT} (h_{in} - h_{in}^*) - Q, \quad (5)
\]

where \(J \text{ [kW]}\) represents the shaft work of the appropriate part, \(F_{IT} \text{ [kg/s]}\) the steam flow, \(h_{in} \text{ [kJ/kg]}\) the enthalpy beyond the throttling valve, \(h_{in}^* \text{ [kJ/kg]}\) the enthalpy at the backpressure and \(Q \text{ [kW]}\) the work of the lost heat.

MODEL OF THE STEAM TURBINE IN MATLAB

The data measured at the real facility should be provided for the accurate model design. They are also used for its validation. The turbine employer granted the current values of the steam flow \(F_{IT}\), the input pressure \(p_{IJ}\) and temperature \(T_{IJ}\), the backpressure \(p_{in}\) and temperature \(T_{in}\) and the actual values of the two-part backpressure turbine power \(J\). Unfortunately, the set of the pressures beyond the throttling valve \(p_{IJ}\) required for the identification was not given. This unavailable data had to be replaced by the appropriate estimation but the executed substitution decreased the design accuracy.

Search for the coefficient \(c\) in Stodola formula

The coefficient \(c\) in Stodola formula relates the steam flow with the pressure difference at the part. Its value is calculated through the real data via simplified equation (2)

\[
c = \frac{F_{IT} \sqrt{T_{in}}}{P_{in}} \left(1 - \frac{P_{in}}{P_{in}^*} \right)^{1/2}, \quad (6)
\]

Gaining the results, it turns out that the parameter \(c\) does not equal for given pressure, temperature and flow to the single value but to the set of quantities (see fig. 2). This can be explicated by the bounded applicability of (2) to the specific range of the pressure difference. Indeed, it is possible to replace the displayed development by the unique value \(c\). It will lead to the bad approximation of the real steam flow by the model. For the compensation of these disproportions, the corresponding \(c\) is assigned to the every value of the pressure difference. Consequently, the real and the simulated steam flow developments cohere.

**Figures 2: Coefficient \(c\) in Dependence on the \((p_{IJ} - p_{in})\)**
Search for the thermodynamic cycle efficiency $\eta$

The thermodynamic cycle efficiency $\eta$ represented by the rate of enthalpies

$$\eta = \frac{h_{h_2} - h_{h_3}}{h_{h_2} - h_{h_3}},$$  \hspace{1cm} (7)

is expressed by the amount of the really transmitted steam enthalpy to the steam enthalpy transmitted in the ideal situation ($\eta = 1$). The search for optimal efficacy consists in the calculus of the output temperature $T_{th}$ for different $\eta$ and in the comparison between gained dependence $T_{th} = f(T_{th})$ and the real one. However, still unknown enthalpy loss $\Delta h$ at the throttling valve considerably influences the executed computation of the $T_{th}$. Hence, the comparison of the developments $T_{th} = f(T_{th})$ is executed only for the higher flows, where $\Delta h \geq 0$ can be assumed. The data mismatch at the low flows is not important at the moment and is tuned later.

The figure below displays the result of the comparison procedure. It can be seen that the thermodynamic cycle of the turbine part has the efficiency $\eta$ about 0.8.

**Figures 3: Finding of the Optimal Efficiency $\eta$**

Search for the optimal enthalpy loss $\Delta h$

The preceding paragraph described the finding of the optimal efficiency $\eta$; the optimal is called because its grow would cause the violence of the thermodynamic theorems ($\Delta h < 0$) and its depression would evoke needless addition of the thermal losses (Nožička 2001). Consequently, the modeled power would be minor than the real one. However, the calculated results do not match the operational data yet (see fig. 3). The last parameter, which has to be set in the model, is represented by the valve characteristic $\Delta h = f(T_{th})$. The manual selection of the enthalpy loss $\Delta h$ at the single steam flow subintervals ensures the sufficient approval of the real and modeled developments.

**Figures 4: Suited Choice of Characteristic $\Delta h = f(T_{th})$**

**Figures 5: Comparison of the Modeled and the Real Data**

The second turbine part is modeled by the same way. The total cascade power consists in the sum of the individual part shaft works.

**Figures 6: Comparison of the Modeled and the Real Power**

Designed model embodies some disproportions against the real equipment. Their mean value is about 7 percent. The accurate measuring of the pressure beyond the valve $P_{ni}$ might minimize these differences; the simulation proved that the used $P_{ni}$ estimation was not too much correct.
TURBINE IN THE SIMULATION TOOL SIMULINK

After determining the unknown parameters of the modeled system, the design is tested in the SIMULINK. The header pressure \( p_{TH} \) and the steam flow \( F_{ST} \) represent the state variables; the turbine part corresponds to the second-order system. Each state variable has an appropriate time constant; \( \tau_P \) responds to the steam delay in the pipes and \( \tau_H \) to the flow integration. The last unknown coefficient \( k \) from (1) describing the relation between the valve position and its pressure loss \( \Delta p \) is identified in accordance with the pressure loss at fully open valve. The turbine part scheme consists of the amount of smaller blocks representing the particular equations (1) – (7).

The enthalpy loss at the valve \( \Delta h \) parameterized by the flow \( F_{ST} \) is created through the two-dimensioned table, where the functional dependence \( \Delta h = f(F_{ST}) \) can be defined. The choice of coefficient \( c \) according to the pressure difference is implemented by the same way. The Steam Properties Package for MATLAB was used for the execution of the entire thermal calculations.

The functions of the appropriate blocks result from their labels. The turbine part was testing in SIMULINK for the different valve position (0 – closed, 1 – fully open), which provides the sufficient information for the evaluation of the model properties and qualities.

The two-part turbine is simulated for different positions of both valves. The input pressure \( p_{ST} \) and temperature \( T_{ST} \) conform to the backpressure \( p_{TH} \) and to the temperature \( T_{H} \). These values with the valve position represent the initial conditions for the second turbine part execution. The simulation is expanded for the situation when the boiler embodied some technical problems and it had to be put aside. The positions of the valves stay the same as in the simulation without the failures.
CONCLUSION

In this paper, a mathematical model for a multipart steam turbine has been designed. After, some model simulations have been done in SIMULINK and the block representing the turbine part has been implemented in this tool. The designed model follows up the real plant trends and the difference between simulated and true data is small. This model accuracy empowers the large usability of the gained design, mainly for the electrical energy control problems in the large-scale energetic systems.

REFERENCES

Ambróž, J. 1980. The steam turbines and the condensation. Faculty of Mechanical Engineering, CTU Prague.

ACKNOWLEDGMENTS

The research has been supported by the grant 102/03/H116 of the Grant Agency of the Czech Republic. I would like to thank Vladimír Havlena for his help with investigated problems.

AUTOR BIOGRAPHY

MILAN CEPÁK was born in Strakonice, Czech Republic and went to the Czech Technical University in Prague, to the Faculty of Electrical Engineering where he studied the system control and obtained his Master degree in 2003. Now, he is a PhD. student at the Department of Control Engineering at CTU in Prague.
QUASI-2D STEAM GENERATOR MODELLING WITH MODELICA

Cécile Avenas  
Daniel Bouskela  
Ugo Schück  
EDF R&D  
6 quai Watier  
78401 Chatou cedex, France  
E-mail : cecile.avenas@edf.fr

Thomas Freude  
Olivier Hascoët  
EDF SEPTEN  
12-14 av. Dutrievoz  
69628 Villeurbanne, France

Nicolas Desvignes  
CRIL TECHNOLOGY  
Le Newton 9-11 rue Jeanne-Braconnier  
92360 Meudon-La-Forêt, France

ABSTRACT

Steam generators are important devices for the operation of pressurized water reactors. Undesirable transients may lead to the automatic shutdown of the plant. To simulate the complex dynamic physical behaviour of the steam generators, a quasi 2-D model called DRIF-BI has been developed using Modelica. Modelica is a declarative physical modeling language normally used for 0-D or 1-D problems. DRIF-BI was previously written in Fortran. The Modelica version has been improved in order to give better predictions and understanding of important standard plant transients such as the loss of one feedwater pump. This work shows that it is possible to go beyond the natural 1-D modeling scope of Modelica, and represent complex two-phase flow phenomena within a large industrial equipment.

INTRODUCTION

The EDF nuclear power plants belong to the pressurized water reactors type (PWR). In PWRs, water acts as the neutron moderator and the heat-transporter fluid. PWRs have two main fluid loops, called primary and secondary fluid loops. The primary water is heated in the core of the reactor and maintained at a high pressure in order to stay liquid. The hot water leaving the reactor vessel enters the U-tubes of the steam generator (SG). The water leaving the U-tubes flows back to the reactor vessel. Secondary water within the SG becomes steam as it receives energy from the primary loop. Steam leaving the SG flows to the turbine which drives the alternator to produce electricity. Steam leaving the turbine passes through the condenser, and then flows back to the SG in the form of liquid feedwater.

Steam generators are key components for operation and also for the safety of the plant because they ensure the cooling of the reactor. Undesirable transients may lead to the automatic shutdown of the plant. That is why it is important to have a clear understanding of the complex dynamics of steam generators, such as the shrink and swell phenomenon of the SG water level. Among important engineering issues related to SG dynamics are the SG water level control or the power dissymmetry between primary loops.

SG physical behaviour

The primary water flows into thousands of U-tubes and becomes cooler as it provides its heat to the secondary fluid, a mixture of water and steam circulating outside the tubes. The feedwater enters the tank and flows down along the down-comers. At the bottom of the down-comers, it enters the central boiling section where it is heated up to boiling, turning into a two-phase flow emulsion. At the top of the boiling section, the circulation rate, ratio between the total flow and the steam flow, is between 4 and 5 at full power. The water/steam mixture passes through the primary steam separators where the two phases are separated. The liquid part is mixed with the feedwater. The steam passes through the secondary separators and leaves the steam generator with a quality of 99.75%.

Why a quasi-2D model?

Modelling the dynamic SG model is difficult because of the complexity of the physical phenomena occurring within the central boiling section. In the case of steady state studies, it is sufficient to model the boiling section as a single 1-D cavity since only global average quantities are needed. Such simple 1-D models are called mono-axial, and are the most commonly in use. However, the boiling section is not an isotropic 1-D cavity: the primary fluid gets cooler while flowing into the U-tubes, which start and end at the bottom of the boiling section, so that the heat flux received by the secondary fluid is not the same on the left and right side of the boiling section. 3D CFD codes give a good representation of this kind of dissymmetry, but they are too detailed and too expensive in CPU time for global plant studies. This is why we have developed an intermediate quality level model for dynamic studies, called DRIF-BI. The need of an intermediate level modelling for dynamic studies has led to the development of DRIF-BI model.

Why modelling with Modelica?

DRIF-BI was originally written in Fortran. We have rewritten DRIF-BI in Modelica to test the capabilities of the
DESCRIPTION OF THE DRIF-BI MODEL

DRIF-BI is a bi-axial model that divides the boiling section into two dissymmetrical legs, the hot leg and the cold leg. It computes the local variable values necessary to account for the observed boiling section dissymmetry. To that end, the two legs exchange mass, heat and momentum. In each leg, the vertical flow is described by a 1-D set of balance equations. The horizontal transfers between the two legs are represented by another 1-D set of balance equations. Hence, the model is locally nearly 2-D in the boiling section, and of intermediate complexity level between 1-D and 3-D models.

DRIF-BI is divided into modules of six different types (see Figure 1). The modules are connected to each other to make a 1-D network that complies with the EDF thermofluid library numerical scheme.

The diagram of the DRIF-BI model, as it has been created in the graphic interface of the supporting tool, is shown in the appendix.

PHYSICS OF THE BOILING SECTION

The vertical flow in each leg is given by the three mass, momentum and energy balance equations for the mixture (or homogeneous) two phase-flow variables (see equations 1, 2 and 3):

\[ \frac{\partial \rho_m}{\partial t} + \frac{\partial}{\partial x} \left( \rho_m V_m \right) = 0 \]  
\[ \frac{\partial \rho_m V_m}{\partial t} + \frac{\partial}{\partial x} \left[ \rho_m V_m^2 + \alpha (1 - \alpha) \frac{\rho_l \rho_v}{\rho_m} V_v^2 \right] + \frac{\partial}{\partial x} P_m - F_m - \rho_m g = 0 \]  
\[ \frac{\partial \rho_m H_m}{\partial t} + \frac{\partial}{\partial x} \left[ \rho_m V_m H_m + \alpha (1 - \alpha) \frac{\rho_l \rho_v}{\rho_m} L V_v \right] - \frac{\partial}{\partial x} \left( \rho_m V_m + \alpha (1 - \alpha) \frac{\rho_l \rho_v}{\rho_m} V_v \right) - \Phi_m = 0 \]

where:
- \( \alpha = \frac{A_v}{A} \) is the void fraction (ratio of volume occupied by vapour on total volume of fluid)
- \( L \) is the difference between vapour and liquid enthalpy \( (H_v - H_l) \)

The three unknowns to be resolved are the pressure \( P_m \), the dynamic enthalpy \( H_m \), and the mass flow rate \( Q_m = \rho_m V_m \). Thus, the model is made up of this 3-equations system and assumes that:
- the two phases are at thermodynamic equilibrium,
- the dynamic imbalance \( V_{d} \), difference between vapour and liquid velocities, is taken into account.

These equations contain unknowns which must be given by so-called closure equations. Closure equations are empirical correlations (obtained from experiments), or additional model equations (here the drift-flux model) (Collier 1972). There are three closure equations:

- the thermal density power exchanged \( \Phi_{m} \), which depends on the thermal flux at the wall of the tubes given by \( \phi_{m} = h(T_{p} - T_{f}) \), where \( T_{p} \) and \( T_{f} \) are respectively the wall and the secondary side temperature. The thermal exchange coefficient \( h \) is given by thermal exchange correlations, that take into account the tube material and the heat exchange process in one or two-phase flow.

- the friction term \( F_{m} \) is given by friction correlations, that depend on geometrical characteristics and on the flow orientation with respect to the tubes orientation.

- the dynamic imbalance \( V_{d} \) is given by a drift-flux model, fitted on specific experiments. Its aim is to account for the important interactions between the two phases, so that the SG flow model is more precise than a simple homogeneous flow model. Zuber and Findlay drift-flux model (Zuber and Findlay 1965) assumes both the existence of a local phase velocity difference \( U_{d} = U_{v} - U_{l} \) and non-uniformity velocity and void fraction profiles.

THE MODELICA LANGUAGE

Modelica (Modelica Association 2002) is a rather new modelling language designed to allow efficient modelling and simulation of complex multi-domain physical systems described by differential, algebraic and discrete equations (DAE). It is a free (non-proprietary) language designed by the non-profit international Modelica Association.

Modelica is a declarative language : the model is given by a set of equations which specifies the physical behaviour of the system. No manual manipulation of the equations is needed in order to build the executable code. This is done by a code generator (Brück et al. 2002), which automatically produces the optimised code suitable for simulation with standard solvers such as DASSL (Petzold 1982), Euler, etc. The code generator performs tasks such as automatic selection of states and index reduction, and causality analysis (i.e. how to solve the algebraic loops). Hence, a lot of computer programming effort can be saved and numerical expertise can be reduced, giving more time to work on the real modelling issues and reducing the overall cost of simulator development.

Modelica is an object-oriented language which encourages a component oriented modelling approach. It is thus possible to design reusable component libraries, and build models graphically by drag and drop from the libraries. A number of free and commercial libraries already exists in many different domains such as mechanics, electronics, hydraulics, electric power, process control, etc.

THE EDF THERMOFLUID LIBRARY NUMERICAL SCHEME

A thermofluid Modelica library is under construction at EDF. Its aim is to provide the framework and the basic and advanced components needed to efficiently build the steady-state and dynamic simulators used for the engineering studies of electrical power plants under normal and incidental operating conditions. The library handles single and two-phase flow, zero and reverse flow, compressible and incompressible flow for water/steam and smoke networks.

Modelica supports DAEs only. To do so, it provides an operator for time derivatives called \texttt{der()}. Some proposals have been made to handle partial differential equations for distributed models (Buchman et al. 2002), but there is no official implementation yet of PDEs within Modelica. Therefore, the discretization equations must be derived by hand from the spatial differential equations. There is no difficulty to write these equations in Modelica.

The EDF thermofluid library uses a finite volume approach, based on the staggered grid scheme for space discretization, and the upwind scheme for the handling of flow reversal (Patankar 1980). Both schemes are well suited for convection, which is the predominant energy transport law within the network. Discretization is performed along the main flow direction only (1-D modelling).

The basic model components are divided into two groups : nodes and edges. Nodes represent mixing volumes such as tanks, boilers, splitters and mergers, etc. They implement the mass and energy balance equations. Edges represent flow resistant elements such as valves, simple pressure loss pipes, etc. They implement the momentum balance equations. The network is built by connecting edges to nodes in order to obtain a complete set of mass, energy and momentum equations with their closure equations (so that the problem is well posed), and automatically fulfil the numerical scheme requirements. Complex library components such as heat exchangers, evaporator pipes or the steam generator described in this paper are also built by assembling edge and node elements. It is also possible to connect edges to edges (e.g. a pipe to a valve) in order to build a network that has a more natural technological appearance, but this may lead to a high index problem if the momentum balance equations are dynamic (a high index problem, meaning that the number of dynamic degrees of freedom is less than the number of dynamic states (Martinson 2000), will never occur when one connects
edges to nodes only). Figure 2 shows the staggered grid for the flow network.

Black rectangles represent mixing volumes. Red rectangles represent flow resistant elements. Pressures \( P \) and mixing enthalpies \( hm \) are computed at the centre of the mixing volumes. Mass flows \( m \dot{\text{c}} \) and enthalpy flows \( h \) are computed at the edges of the mixing volumes (or equivalently at the centre of flow resistant elements). \( P, hm \) and \( m \dot{\text{c}} \) are computed by the mass, energy and momentum balance equations. The upwind scheme states that the enthalpy flow \( h \) is equal to the upstream mixing enthalpy \( hm \). Arrows give the positive flow direction.

The model network (see Figure 3) is built by connecting volumes and flow resistant elements in alternating sequences. Information between elements are exchanged through the connectors C.

**Figure 2 : Flow network**

**Figure 3 : Model network**

**SIMULATION RESULTS**

DRIF-BI is used to perform plant studies under normal, incidental or accidental operating conditions. To that end, scenarios are fed as input data into the model. The aim of these studies is to understand how important parameters such as the SG water level, the primary loop temperature and the dome pressure react to these scenarios.

Under normal or incidental operating conditions, steps are given as boundary conditions for the feed-water flow, the output steam flow and the primary water energy. These simulation cases are especially useful to study the impact of SG replacement. It is also possible to feed as input data the events as measured on-site.

In this way, it has been possible to reproduce the SG behavior leading to the automatic shutdown of the plant due to the loss of the main feed-water pump.

The examples below are some results of feed-water pump loss simulations with Dymola using the DASSL solver. Figure 4 shows the site measurements data fed into the Modelica model as boundary conditions. Figure 5 shows the dome pressure calculated by Dymola. Figure 6 shows the shrink and swell phenomenon of the water level as computed with Dymola and as measured on-site.
CONCLUSION

A quasi-2D model of a steam generator has been developed using the new modelling language Modelica with the supporting tool Dymola. The model gives a good representation of the two-phase flow physics within the steam generator. With Modelica, it was relatively easy to improve the physics of the model in order to take into account dynamic terms that were previously neglected in the Fortran version. Thus, an even more predictive model has been achieved. The model in Modelica is also more readable, giving more time to work on the real modelling issues.

This model predicts the automatic shutdown of the reactor due to the loss of one feed-water pump. In the future, it could be used to study the effects of excessive grid load, and the effects of the reactor rod ejection.

This work shows also that it is possible to go beyond the natural 0-D/1-D scope of the Modelica language.

Other industrial cases, such as the modelling of a whole plant, are under investigation at EDF to decide whether Modelica is a good successor for traditional Fortran modelling of these types of problems.

ACKNOWLEDGEMENTS

The development of the Modelica models is a collaborative effort. Many thanks to Annick Souyri, Jean-Daniel Mattei, Franck David, and Emmanuelle Petit for giving fruitful comments and feedback.

REFERENCES


ENERGY CONTROL SYSTEMS SIMULATION
APPLICATION OF MULTI-AGENT TECHNOLOGY TO FAULT DIAGNOSIS OF POWER DISTRIBUTION SYSTEMS

Jusong Yang, Brian Carse, Mohammad Montakhab*, Anthony G. Pipe and Terence S. Davies
Faculty of Computing, Engineering and Mathematical Sciences, University of the West of England, Coldharbour Lane, Bristol BS16 1QY, United Kingdom
*Corresponding Author. Tel. +44-117-3282951, Fax. +44-117-3283636   Email: Mohammad.Montakhab@uwe.ac.uk

ABSTRACT

When a fault occurs in a power system, the protective relays detect the fault and trip appropriate circuit breakers, which isolate the affected equipment from the rest of the power system. Fault diagnosis of power systems is the process of identifying faulty components and/or sections by analysing observable symptoms (telemetry messages). As the domain itself is characterised by dynamic situations, extensive telemetering, complex operations, and distribution of lines and substations over a large geographical area, it is difficult to tackle fault diagnosis problems through the strength and capability of a single intelligent system. This paper describes an experimental multi-agent system developed for and aimed at a computer-supported fault diagnosis in electricity distribution networks. Some example faults have been simulated on the distribution network. The system is based on a hierarchy of five agents that co-operate with each other in order to diagnose a fault.

Keywords: software, fault diagnosis, multi-agent systems, simulation, expert systems, decision making, decision support.

1 INTRODUCTION

The electricity distribution industry is one of the largest in the UK, serving nearly 27 million customers. In England and Wales, there are 12 regional electricity companies responsible for distributing and supplying electricity to customers. Electricity is received from the transmission network at 132kV and is transformed down to lower voltages, i.e. 33kV, 11kV and 240V, as it is distributed across a region. 132kV and 33kV are referred to as High Voltage (HV) and 11kV and below as Low Voltage (LV). Domestic customers receive their supply at 240V and some large industrial customers can take their supply at either 11kV or 33kV.

On the high voltage networks (132kV, 33kV), the circuits are usually equipped with automatic circuit breakers that report their operation to the control centre via the Supervisory Control and Data Acquisition (SCADA) system. These circuit breakers are fitted with an auto recloser, this means that when they open automatically due to a fault, after a short delay they will reclose in an attempt to restore the supply. If the fault is transient, the circuit breaker will remain closed and the power will be restored, but if it is permanent then the circuit breaker will reopen again to disconnect the supply.

With the low voltage networks (11kV and below), often only outgoing source circuit breakers, on 11kV feeders from major substations, are telemetered. Below the source circuit breaker the network is radial with many branches that lead to the 11kV/415V transformers supplying the customers. These branches are protected by protective devices, such as fuses, which are not telemetered.

A control engineer who works in the control centre directly controls the network, based upon indications and alarms available to him/her and through the use of telecontrol command schemes. When a fault occurs in the network, no direct information is obtained on what type of fault has occurred and where it is. Rather, telemetry messages obtained in the control centre indicate which switches and automatic protective relays have operated in response to the fault. These telemetry messages are normally regarded as ‘post-fault symptoms’, and they are vital information to the control engineer when diagnosing faults.

The task of fault diagnosis would be easier if all the relevant switches and protective relays operated correctly. However, fault diagnosis could become quite complex due to factors such as an item of equipment failing to operate when required, or operating when not required. There are other factors, the telemetry messages received are not always in relation to the time the fault occurred, and there is often simply a lack of detailed information regarding the low voltage networks. In some extreme cases (e.g. blizzards, storms etc.) where several faults occur simultaneously, telemetry messages could arrive at the rate of 2,000 per hour or higher (Brailsford and Cross, 1989). The use of computer-aided systems in fault diagnosis of electricity distribution networks, based on the simulation of the faults, is becoming more and more important to increase effectiveness and efficiency. This paper reports on an experimental research work carried out at Power Systems and Electronics Research Group, University of the West of England, UK. The aim of the project is to develop a multi-agent system for fault diagnosis of electricity distribution networks (Yang, 2001).
2 Traditional AI Approaches

Since the birth of artificial intelligence (AI) based on symbol manipulation, impressive progress has been made in human understanding of the basic issues related to knowledge representation. This has motivated active researchers to apply AI techniques to fault diagnosis of power systems. Many methods have been developed to assist the control engineer in diagnosing faults using expert systems (Fukui and Kawakami, 1986), (Montakhab and Adams, 1998), (Cockburn, 1992), (Wang and Dillon, 1992) and neural networks (Mahamed and Mazumder, 1999), (Swarup and Chandrasekhariah, 1994), (Yang et al., 1994). A common method is an expert systems (ES) approach. The ES approach is appropriate for this task because diagnosis requires the experiential knowledge of an expert, and logic reasoning is the primary task. A good review of the expert systems-based approach to fault diagnosis of power systems can be found in (Sekine et al., 1992). Although good results have been obtained, as they are highly centralised, these systems suffer from some drawbacks as follows:

i. The complexity of an expert system increases rapidly as the size of its knowledge base increases, maintenance of such a large knowledge base often proves to be difficult, and design of these systems often requires considerable efforts because of the complex knowledge acquisition process.

ii. Touching any part of a centralised system often endangers the entire structure, making it difficult to modify and debug.

iii. Large knowledge bases execute less efficiently, because more time is spent in search.

iv. A single point of failure often makes the overall system useless.

v. A centralised system is expensive.

3 Benefits of Multi-Agent Approach

Historically, multi-agent systems technology was invented as a sub-field of distributed artificial intelligence, which itself is a sub-area of artificial intelligence. Today, the term ‘multi-agent systems’ is used to refer to all types of systems composed of multiple (semi) autonomous components (Jennings et al., 1998). This approach represents a new and promising solution to problems outlined above.

In multi-agent systems, individual problem-solving entities are called agents; agents are grouped together to form communities which co-operate to achieve the goals of individuals and of the system as a whole. It is assumed that each agent is capable of a range of useful problem-solving activities in its own right, has its own aims and objectives and can communicate with others (Jennings et al., 1993).

As distributed systems, multi-agent architectures have the capacity to offer several desirable properties over centralised systems (Stone and Veloso, 2000):

**Speed-up and Efficiency.** Agents can operate asynchronously and in parallel, and this can result in an increased overall speed.

**Robustness and Reliability.** Each agent is more reliable because of its reduced complexity. The failure of one or more agents does not necessarily make the overall system useless, because the control and responsibilities are shared among different agents.

**Scalability and Flexibility.** The system can be adapted to an increased problem size by adding new agents, and this does not necessarily affect the operationality of the other agents.

**Costs.** Since it could be composed of simple subsystems of low unit cost, it may be much more cost-effective than a centralised system.

**Development and Reusability.** Individual agents can be developed separately, either from scratch, or on the basis of already available hardware and/or software facilities. The overall system can be tested more easily, and it may be possible to reconfigure and reuse agents in different application scenarios.

4 MAFS: A MULTI-AGENT FAULT DIAGNOSIS SYSTEM

In this section we describe the approach underlying the Multi-Agent Fault Diagnosis System (MAFS).

4.1 MAFS Architecture

In the architecture shown in Figure 1, fault diagnosis of power systems is devolved to five agents, a Power System Agent (PSA), a High Voltage Diagnosis Agent (HVIDA), a Substation Diagnosis Agent (SDA), a Low Voltage Diagnosis Agent (LVDA) and a Global Decision Agent (GDA). The arrows indicate the inter-agent communication.
4.2 Role of Each Agent

PSA -- The function of this agent is to report fault symptoms associated with a fault to the HVDA, SDA and LVDA as required. The fault symptoms are acquired based on the simulation of the faults on the different parts of the power system. In fact, PSA is nothing more than a simulator.

HVDA -- The function of this agent is to diagnose the location and type (permanent or transient) of faults on the high voltage circuits (132kV and 33kV).

SDA -- The function of this agent is to diagnose the location and type of faults on substations of the distribution network (132/33kV and 33/11kV).

LVDA -- The function of this agent is to diagnose the location and type of faults on the low voltage networks (11kV and below).

GDA -- The function of this agent is to make a final diagnosis based on the local hypotheses from HVDA, SDA and LVDA.

In fact, HVDA, SDA and LVDA simulate different faults in their own part of the network and try to find the best match.

4.3 Agent Interactions

There are 2 categories of agent interactions:

4.3.1 Interaction with Control Engineer
The human control engineer is able to enter events into agents’ databases. In addition, agents are able to interrogate the control engineer for additional information in the case of incomplete data or partial hypotheses.

4.3.2 Inter-Agent Communication
All agents are able to request information from the PSA telemetry database. HVDA, SDA and LVDA exchange fault diagnosis hypotheses with GDA in order to reach a final diagnosis. Inter-agent communication is achieved using KQML as described below.

4.4 Protocol Language

The notion of a multi-agent system presumes that individual agents can exchange meaningful information in order to achieve individual goals and a shared global goal. A protocol language provides a semantically rich level of discourse for agent communication and defines the structure of the messages so that they are understandable to all agents. Various agent protocol languages have been devised. These include KQML (described below) and the FIPA (Foundation for Intelligent Physical Agents) Agent Communication Language (ACL) (O'Brien and Nicol, 1998). In its initial implementation, MAFS employs part of the KQML language. Future developments in MAFS will involve replacing KQML with the FIPA ACL since the latter is more formally defined while still retaining the usability of KQML; a detailed comparison of KQML and FIPA ACL is given in (Chaib-draa and Dignum, 2002).

The Knowledge Query and Manipulation Language (KQML) is a protocol language that is designed to support interactions among distributed intelligent agents (Finn et al., 1997). It provides a standard message format for knowledge sharing between agents.

When using KQML, a software agent composes messages in its own representation language, wrapped in a KQML message, and relies on a separate router process to deliver KQML messages across the network to the agent with whom it wishes to communicate (Figure 2).

![Figure 2](image)

4.5 Communication Mechanism

While KQML defines the structure of messages exchanged by agents, the question now is how a message from one agent to another agent can be delivered. In other words, a communication mechanism is needed to carry the message across the network to its final destination.

In MAFS, inter-agent communication is achieved using sockets – program-defined end-points for network communication between processes. The underlying transport protocol here used is TCP/IP.

5 CASE STUDIES

MAFS has been tested on a number of fault scenarios, using model distribution networks. This section presents three case studies of increasing complexity. In each case, a linguistic interpretation of the multi-agent system fault diagnosis is given.

5.1 Case Study #1

The fault for Case Study #1 is shown in Figure 3. At 13:30 hours, the transformer T12 had faulty contacts. Consequently the Buchholz relay, which protects T12, operated an alarm.

5.1.1 Fault Diagnosis

**Stage 1**

HVDA asks PSA about local telemetry, and the PSA informs the HVDA that there are no telemetry messages.
SDA asks PSA about local telemetries, and the PSA informs the SDA that at 13.30 hours, the Buchholz relay of the transformer T12 operated an alarm.

LVDA asks PSA about local telemetries, and the PSA informs the LVDA that there are no telemetry messages.

**Stage 2**
HVDA initiates local diagnosis. Because there are no local telemetry messages, the HVDA makes the hypothesis that there are no local faults.

![Diagram](image)

**Figure 3**

SDA initiates local diagnosis. After analysing the telemetry messages, the SDA makes the hypothesis that a minor fault may have occurred in transformer T12.

LVDA initiates local diagnosis. Because there are no local telemetry messages and no telephone calls from customers reporting loss of supply, the LVDA makes the hypothesis that there are no local faults.

**Stage 3**
GDA compares the local reports from HVDA, SDA and LVDA. Since the local results from HVDA and LVDA are negative, the conclusion from the GDA is the same as the one made by SDA.

**5.2 Case Study #2**

The network and fault for Case Study #2 is shown in Figure 4. At 16.00 hours a lightning strike hit phase A of the above 11kV line causing a phase to ground fault on phase A. Consequently the instantaneous overcurrent relay of CB18 operated, but due to mechanical problem CB18 failed to operate. As a result, standby earth fault relays of transformers T9 and T10 operated and CB13 and CB15 opened to isolate the faulty section from the rest of distribution network. The first “no supply” calls from postcode PC1 were received after 12 minutes.

**5.2.1 Fault Diagnosis**

**Stage 1**
HVDA asks PSA about local telemetries, and the PSA informs the HVDA that there are no telemetry messages.

SDA asks PSA about local telemetries, and the PSA informs the SDA that at 16.00 hours, standby earth fault relays of transformers T9 & T10 operated and circuit breakers CB13 and CB15 opened.

LVDA asks PSA about local telemetries, and the PSA informs the LVDA that at 16.00 hours, instantaneous overcurrent relay of circuit breaker CB18 operated and the load flows through 11kV circuits CCT1 & CCT2 were reduced to zero.

**Stage 2**
HVDA initiates local diagnosis. Because there are no local telemetry messages, the HVDA makes the hypothesis that there are no local faults.

SDA initiates local diagnosis. Since the circuit breakers CB13 & CB15 opened as a result of operation of the standby earth fault relays, which provide earth fault protection for busbar BB1, the SDA makes the hypothesis that there may be an earth fault on busbar BB1.

LVDA initiates local diagnosis. By using the information related to the interrupted customers (postcode, time of call), the LVDA infers that the fault is related to one of the 11kV circuits CCT1 and CCT2 and one of the secondary transformers T1, T2 and T5. The LVDA asks the user whether there are any pre-known damaged plants. The answer is negative. The LVDA also wants to know if there is a sudden load reduction in CCT1 and CCT2. The answer is that they both have a 100 percent change of load. The LVDA also asks the user whether there are lightning strike reports at 16.00 hours. The answer is positive. The x and y co-ordinates of the lightning point is 200 and 373. The LVDA takes this piece of information to calculate the distance between the lightning point and above suspected circuits. If the distance is less than 3km, then there is possibility that the line has been damaged. In this case, the distance between lightning point and CCT2 is 500m, so it is highly possible that the lightning has damaged CCT2. In addition, telemetry messages show that the relay of circuit breaker CB18 has operated, but circuit breaker CB18 itself has not. In the end, the LVDA makes a hypothesis that circuit CCT2 may have been damaged by lightning strike and circuit breaker CB18 has malfunctioned and circuits CCT1 and CCT2 are off supply.

In this case, four more phone calls from postcode PC2 and PC3 were received following the first one. As LVDA has already diagnosed the fault, the LVDA takes no further action.

**Stage 3**
GDA compares the local reports from HVDA, SDA and LVDA. Both of the SDA and the LVDA report possible local fault in its own region. In order to draw any conclusion, the GDA needs to identify whether there are any possible connections between them. In this case, the GDA tries to relate possible fault identified by the SDA to the malfunction and the fault reported by the LVDA. In the end, the GDA concludes that the actual fault is the one identified by the LVDA.
5.3 Case Study #3

At 20.40 hours, a low impedance fault occurred in the high voltage winding of the primary transformer T10 as shown in Figure 5. Consequently the Buchholz relay that protects T10 operated and sent trip signals to circuit breakers CB14 and CB15. Due to mechanical problem CB14 failed to open, but CB15 opened correctly. As a result, backup overcurrent protection on the grid transformers T15 & T16 operated and circuit breakers CB3 & CB5 opened to isolate the faulty component T10 from the rest of distribution network. As primary substations SS2, SS3 and SS4 lost their supply, customers supplied by the above substations were all affected.

5.3.1 Fault Diagnosis

Stage 1
HVDA asks PSA about local telemetry messages, and the PSA informs the HVDA that at 20.40 hours, the load flows through 33kV lines L1–L6 was reduced to zero.

SDA asks PSA about local telemetry messages, and the PSA informs the SDA that at 20.40 hours, the Buchholz relay of Transformer T10 operated and circuit breaker CB15 opened, and backup over-current protection on the grid transformers T15 & T16 operated and circuit breakers CB3 & CB5 opened.

LVDA asks PSA about local telemetry messages, and the PSA informs the LVDA that at 20.40 hours, the load flows through 11kV lines CCT1–CCT8 were reduced to zero.

Stage 2
HVDA initiates local diagnosis. The telemetry messages show that there are no supplies in lines L1–L6. However, there are no related relay or circuit breaker operations reported. Based on the above information, the HVDA makes the hypothesis that lines L1–L6 are off supply and there are no local explanations.

SDA initiates local diagnosis. After analysing the telemetry messages, the SDA infers that transformer T10 may be faulty (since its Buchholz relay and circuit breaker CB15 has operated) and circuit breaker CB14 has malfunctioned (since the Buchholz relay that sends trip signal to CB14 has operated as well as its backup overcurrent protections on transformers T15 & T16). In the end, the SDA makes the hypothesis that a fault may have occurred on transformer T10 and circuit breaker CB15 has opened, but CB14 has malfunctioned. The backup overcurrent protections on transformers T15 & T16 have operated and consequently circuit breakers CB3 & CB5 have opened.

LVDA initiates local diagnosis. The telemetry messages indicate that there are no supplies on the circuits CCT1–CCT8, but no indication of any operation of relays or circuit breakers related to the above circuits. However, many customers reporting loss of supply from postcodes PC1–PC9 were received. On the basis of the above information, the LVDA makes the hypothesis that circuits CCT1–CCT8 are off supply and there are no local explanations.

Stage 3
GDA compares the local reports from HVDA, SDA and LVDA. Both of the HVDA and the LVDA report loss of local supply and no local explanations were obtained. However, the SDA reports that transformer T10 may be faulty, one of its circuit breakers has malfunctioned, and backup overcurrent protections on the grid transformers have operated. In this case, the GDA tries to relate the loss of supplies of HVDA and LVDA to the possible fault identified by the SDA. In the end, the GDA concludes that the actual fault is the one reported by the SDA.

6 CONCLUSION AND FUTURE PROSPECTS

Experiments have shown that a multi-agent system approach can serve as a good formalism for fault diagnosis of electricity distribution networks, and with a good potential for real-time application in terms of speed, reliability, robustness and costs. Moreover, the system’s modular structure of autonomous agents makes it easily modified. As the network changes, new agents can be added in with minimum disruption to other agents. The software developed can effectively help the control engineers in decision making. This decision support can be invaluable during the major system disturbances, during which many faults are generated simultaneously and the control engineer is overwhelmed by a large number of generated alarms.

7 ACKNOWLEDGEMENTS

The authors wish to express their thanks to Mr. Neil Griffiths from Western Power Distribution, for his assistance in preparation of case studies presented in this paper.

8 REFERENCES


Symbol used:
- Closed circuit breakers
- Tripped circuit breakers
- Relay operated normally, circuit breaker failed to operate

Figures 4 & 5
MODELS AND METHODS OF DESIGNING THE INTELLIGENT DECISION
SUPPORT SYSTEMS FOR MANAGING A POWER BLOCK

Vadim N. Vagin and Alexander P. Eremeev
Department of Applied Mathematics
Moscow Power Engineering Institute (Technical University)
Krasnokazarmennaya, 14, 111250, Moscow,
Russia
E-mail: vagn@apmsun.mpei.ac.ru
ereeeevv@apmsun.mpei.ac.ru

KEYWORDS
Decision support system, expert system model, parallel methods, dynamic model

ABSTRACT

This paper describes designing the real time decision support system (RTDSS) of a semiotic type. RTDSS of such type is oriented towards open and dynamic problem domains and it is defined by the set of various models. The example of a RTDSS prototype for monitoring and management of a nuclear power block implemented on the basis of the tools G2+GDA is viewed. This work was supported by the RFBR (project 02-07-90042)

INTRODUCTION

Real time decision support systems (RTDSS) are hardware-software complexes, intended for the help to the decision making persons (DMP) at the management of complex objects and processes of a various nature in conditions of rigid temporary restrictions. When searching the decisions, expert models, constructed on the basis of expert knowledge and heuristic methods of decision searching, are used. According to a modern classification of software, RTDSS are a class of integrated intelligent (expert) systems of a semiotic type, combining exact mathematical decision search methods with inexact, heuristic methods, based on expert knowledge (Vagin et al. 2001; Vagin et al. 1999; Yeremeyev 1997; Eremeev et al. 2002; Vagin et al. 1996; Vagin 1995).

At implementation of intelligent RTDSS the necessity of search of an acceptable decision in conditions of rigid temporary restrictions is displayed. It can be successfully implemented only by the use of hardware-software tools of parallel processing of information. Both at the level of external parallelism (problems, modules, procedures) and at the level of internal parallelism (production rules, operators) such tools were used at construction of the RTDSS prototype for monitoring and management of some systems (the main circulating pump (MCP), the capacitor plant (CAP), the ejector plant (EJP), the subsystem of cooling the reactor zone (SCRZ)) of a nuclear power block, implemented on the basis of the high-effective tool complex G2+GDA (G2-Diagnostic Assistant) at the Applied Mathematics Department of the Moscow Power Engineering Institute (Technical University). Complex G2+GDA operates on a dual-processor workstation SUN Sparc Server 1000 and IBM Pentium computers being used as the working terminals.

RTDSS is based on a knowledge representation model of a semiotic type. The semiotic model for RTDSS is defined by the collection

\[ SS = \{M, R(M), F(M), F(SS)\} \]

where

- \( M = \{M_i: \ldots, M_n\} \) is the set of formal or logic-linguistic models, implementing defined intelligent functions;
- \( R(M) \) is the function for selection of the necessary model in a current situation;
- \( F(M) = \{F(M_1), \ldots, F(M_n)\} \) is the set of modification functions of models \( M_1, \ldots, M_n \);
- \( F(SS) \) is the function for modification of \( SS \) system, i.e. its base components \( M, R(M), F(M) \).

We consider that the knowledge representation model is adequate to a problem domain (semantic correctness) if any state from an acceptable initial state set \( S_i \) is translated into a state from a goal state set \( S_g \) and the process of state conversion is finite.

PECULARITIES OF RTDSS

Peculiarities of problems, solved by RTDSS, are:

- the necessity of the temporary factor account at the problem situation description and during decision search;
- the necessity of decision making in conditions of temporary restrictions, determined by a real controlled process;
- the impossibility of obtaining all objective information, necessary for the decision, and, in this connection, the use of subjective, expert information;
- the complexity of a search, the necessity of an active participation of DMP;
- the presence of nondeterminism, the necessity of a correction and an introduction of additional information during decision search.

A basic purpose of RTDSS is to help to DMP at the control of complex objects and processes, revealing and preventioning of dangers, development of the recommendations, i.e. to help in the sanction of problem situations before they will become irreversible.

The main design principles of RTDSS are:

- openness and dynamics;
- adaptivity and learning;
- semiotics;
- distributivity and parallelism in information processing;
• application of a high performanced computer technique and efficient tools (complete environments) of the type G2+GDA;
• the application of cognitive graphics and a hypertext in information mapping.

The developed RTDSS for operative-dispatching monitoring and management of a nuclear station power block and, in particular, its subsystems – MCP, CAP, EJ, SCRZ, are the intelligent systems of a semiotic type (Vagin et al. 1999).

THE ARCHITECTURE OF RTDSS

The generalized architecture RTDSS is given in Fig. 1. In contrast with traditional expert systems in RTDSS it is necessary to include an additional modeling block and a forecasting block for analysis and estimation of accepted decision consequences and choice of the best recommendations. These blocks are implemented on the basis of imitative modeling system G2.

We shall briefly view the basic concepts of RTDSS base modules implemented on the example of the prototype for monitoring and management of MCP and other subsystems of a power block.

Data base (DB) is a storehouse of object management information, acting with gauges from a subsystem of modeling and from DMP. The data in DB represent a set of continuous and discrete parameters of MCP (CAP, EJ, SCRZ). For database representation, the class hierarchy is implemented. The class definition is given by means of the special tables, including such attributes as: a name of a class (Class name), a name of a parent class (Superior class), description of class specific attributes (Class specific attributes). A base class of MCP parameters is a class of gcn-parameter – a successor of a class “object”. Each MCP parameter has a name represented by DMP, what is reflected by attribute “fullname”. All parameters are displayed as icons. The color of an icon of continuous parameters is changed in depending on a value of a current parameter. At a normal condition the icon color is green, at an abnormal condition – the red color.

The change of color occurs according to production rules as “whenever” at obtaining the value by attribute “state”, for example:

whenever the state of any continuous-parameter CP receives a value and when not (the state of CP is normal) then change the body icon-color of CP to red.

The value is also given to attribute “state” on a rule “whenever” at obtaining of a current value of a continuous parameter. For example:

whenever the cur-value of any continuous-parameter CP receives a value and when the cur-value of CP > the top-set of CP then conclude that the state of CP is upper.

DMP can change the current value of any parameter, if for any reasons a source of parameter data (gauge) has ceased to function.

Knowledge base (KB) contains expert knowledge, on the basis of which analysis of object state will be carried out. The kind, in which knowledge in KB is submitted, is determined by a knowledge representation model. The greatest spread in intelligent systems has received a production model with rules of the type:

If <antecedent> then <consequent1> /, else <consequent2>.

In RTDSS an antecedent represents a logical expression concerning parameters of an object state. A consequent describes some action, which should be undertaken in case of the validity (consequent1) or not validity (consequent2) of antecedent. These actions consist in a distribution of commands to the information block, in a change of information about the object state, in activation of some knowledge from KB. The state estimation and the result distribution is a total of such action sequence. KB of RTDSS should satisfy to a number of the requirements, among which: presence of knowledge, allowing to carry out the state analysis of an object at incomplete information; presence for DMP of opportunity of updating KB in a working mode, and, hence, presence of advanced means of the checking up entered data on correctness, as the faulty changes in KB at functioning an object are inadmissible.

The instruction block directs actions of DMP in planned transitive modes. It works automatically (on a situation) at switching on the appropriate mode. The information on a mode of functioning of object acts from DB. The above requirements cause a necessity of information representation in KB in the most convenient for DMP recognition a graphic form.

![Diagram of RTDSS architecture](attachment:image.png)

**Figure 1:** The generalized architecture of RTDSS

The solver is a procedure that implements the algorithms (strategies) of application of knowledge from KB to data from DB. The solver as well as KB is a component of decision making model.

Under designing RTDSS for monitoring and controlling the sophisticated objects, in particular, the semiotic systems of operative-dispatching control by nuclear stations, traditional
methods of information search are impossible in principle
due to huge amount of interrelated and sometimes
inconsistent facts and laws describing these objects.
Therefore for manipulating and processing knowledge,
representing by a collection of production rules, it is
necessary to have an inference engine which implements
decision-making procedures and forms a control
recommendation list for such systems.
We have focused on parallel inference methods and
procedures because only parallelism of inference procedures
allows to increase the speed of inference processes and to
treat a huge amount of productions.
The prototype of RTDSS allows the parallel control for the systems – MCP, CAP, EJ, SCRZ.
The following levels of external and internal parallelism are
implemented.

Parallelism at the level of development. All systems of the
nuclear power block implemented in the prototype are being programmed and processed in parallel.

Parallelism at the level of problems, program modules
and procedures. A few modules carrying out certain
functions can work simultaneously within one application.
At parallel fulfillment of some problems one can arise the
necessity of data exchange and accordingly a problem of
data synchronization. For organization of synchronous
interaction between blocks of various subsystems the
production rules as whenever and when are used. For
parallel processing inside procedures and methods the
construction do in parallel is applied.

Parallelism at the level of rules. The current version of
RTDSS is based on classical production (rule-based) model
of knowledge representation in the most cases. This model is
defined by the collection \( <S,P,ST> \), where: S is the set of
possible states (universe) of a problem domain; P is the
finite set of production rules (productions) implementing the
function P: S → S, and each production \( P \in P \) is the function \( P_i: S → S \).
S ⊆ S; ST is the search strategy set defining the
executable production subset \( \{P\} \subseteq P \) for some current state \( s \) from S. Usually forward or backward search strategies are
used.
It is very important to decrease a search time within the
framework of a design of RTDSS. For achievement of this
goal a parallel organization of the production model is
used.
The condition of parallel executing a production pair \( P_i \) and
\( P_j \) from an activable set is their commutativity: \( P_i \circ P_j \Leftrightarrow P_j \circ P_i \)
COM \( P \), where COM means a commutativity relation
(Vagin et al. 1996; Oshiasano et al. 1987; Moldovan 1986).
Let us define production as follows: \( P_i: L_i → N_i \), where \( L_i \)
and \( N_i \) – some kind of structures and \( K_i = L_i \cap N_i \).
Productions \( P_i: L_i → N_i \) and \( P_j: L_j → N_j \) can be independent
(\( P_i \circ P_j \)) or connected by the following types of dependencies:
on input (\( P_i \circ N_i \)), on output (\( P_i \circ O_i \)), on input-output (\( P_i \circ I_i \) and on interface (\( P_i \circ I_i \)).
Formally we can define it in such a way:

- \( P_i \circ N_i \Leftrightarrow (L_i \cap L_j = \emptyset) \wedge (N_i \cap N_j = \emptyset) \wedge
( L_i \cap N_j = \emptyset) \wedge (L_i \cap L_j = \emptyset) \cup (L_i \cap N_j = \emptyset) \); 
- \( P_i \circ IN \Leftrightarrow (L_i \cap L_j = \emptyset) \wedge (N_i \cap N_j = \emptyset) \); 
- \( P_i \circ OUT \Leftrightarrow (N_i \cap N_j = \emptyset) \wedge (L_i \cap L_j = \emptyset) \); 
- \( P_i \circ IO \Leftrightarrow (L_i \cap L_j = \emptyset) \); 

The following assertions defining a possibility and an
impossibility of parallel executing \( P_i \) and \( P_j \) are proved:

1) \( (P_i \circ P_j) \Leftrightarrow (P_i \circ P_j) \wedge ((P_i, OUT P_j) \lor (P_j, OUT P_i)) \wedge
\neg((P_i, IN P_j) \lor (P_j, IO P_i)) \); 
2) \( \neg(P_i \circ P_j) \Leftrightarrow (P_i, IN P_j) \lor (P_j, IO P_i) \).

At a containable level, assertion 1) means that a parallel
execution of \( P_i \) and \( P_j \) is possible under their full
independence or their dependence on output or on interface.
If \( P_i \) and \( P_j \) are dependent on input or output then
according to assertion 2), their parallel execution is
impossible.

On the basis of these assertions, there is built a parallelism
matrix \( Q = [Q_{ij}] \), \( i,j = 1,2,\ldots,n \), where \( n \) is a production
number in a production system:

\[
Q_{ij} = \begin{cases} 
1, \text{if}(P_i \circ IN P_j) \lor (P_j \circ IO P_i); \\
0, \text{otherwise}. 
\end{cases}
\]

At executing a production program for every production \( P \)
from an activable set, a parallel subset \( R_i \) is defined:

\( R_i = \{P_i\} \cup \{P_j : Q_{ij} = 0\} \).

One can concurrently perform only those productions from
an activable set that occur in an intersection \( \bigcap R_i \).

The parallel processing algorithm of a production system
includes the following stages.

1. Preprocessing treatment:
   - building the parallelism matrix \( Q \),
   - constructing for all productions \( P \), a production system
     the parallel subsets \( R_i \), \( i=1,2,\ldots,n \). 

2. Initialization:
   - the input of an initial state \( s_m \) and setting up a current state
     \( s=s_{mp} \),
   - the input of a goal state \( s_g \), if \( s=s_g \), then stop.

3. Decision search (the main production cycle):
   - computing an activable set for a current state \( s \) (i.e.
     performing a matching phase of a production cycle), defining
     a subset of parallel executable productions \( \bigcap R_i \) being in the
     activable set,
   - performing the productions from \( \bigcap R_i \) and obtaining a
     new current state \( s \); if \( s=s_g \), then stop, otherwise returning
to the beginning of stage 3.

The given model is dynamic and is adjusted to the current
production system: the necessary number of units of each type
is defined, the connections between units are established, etc.

Opportunities of parallel processing of rules initially,
unconditionally, whenever, when, if, for are used.

Parallelism inside rules. For parallel processing inside
rules, i.e. for organization of fulfillment of certain actions
for the whole set of objects of the same type, the generalized
rules on the basis of constructions for and every are applied.
Thus the same process is made active in parallel for the
whole set of objects.

For organization of logic and arithmetic operations and their
parallel processing (level of internal parallelism) blocks
AND, OR, N TRUE (voting), synchronization and hold
are used.
Sometimes the parallel processing is inadmissible, as the actions of rules can depend on each other and the fulfillment of one of them should precede fulfillment other. For example, it is necessary to change any parameter during imitation or forecasting of a real process, and then again to return in an initial situation. For such cases a construction in order is used, with the help of which the order of fulfillment of certain actions is established.

For checking of a condition of a continuous parameter at its forecasting a simultaneous check is used, whether its urgent value above or below than established borders is. The parallel processing of input data allows minimizing the result calculation time.

The forecasting block carries out functions of forecasting of abnormal situations and consequences of managing actions. The forecasting is made on a DMP command on the basis of data on a current state of controlled object, acting from DB, and on the basis of knowledge, stored in KB. The managing action is either a set by DMP, or the recommendation given by G2 solver.

The information mapping block carries out functions of information representation to DMP. Initial data for it are data from DB, results of an estimation of an object state, received by a solver, results of the forecasts, made by the forecasting block, and instructions, given out by the instruction block.

The information, on the one hand, should be displayed in the form convenient for a fast recognition by DMP and, on the other hand, should be as possible more complete. These requirements contradict each other, as the increasing of volume of the sign information decreases an ability of the person to perceive it. The problem is decided by means of the multilevel circuit of information mapping with application of a hypertext technology and cognitive graphics.

Modeling block simulates behavior of an object. It can act as the agent of data about an object state at a stage of system testing and at a decision making stage for comparison with data, acting from gauges. Modeling block can also be used together with the forecasting block for the forecast of abnormal situations and consequences of managing actions.

COGNITIVE GRAPHICS IN RTDSS

The above requirements cause the necessity of the information representation in a knowledge base (KB) in the most convenient for DMP recognition a graphic form using the hypertext technology.

For mapping the information a number of working spaces is entered.

1. A working space with the scheme of subsystems of a power block concerns just to such class, its auxiliary systems and gauges. A subsystem is presented by an icon of a class. The gauges are represented on groups by graphic images of parameters fixed by the subsystem. The working space of parameters becomes visible at pressing of a mouse key on a parameter icon. Dynamics of processes, occurring in a subsystem, is displayed by change of a graphic image color of parameters. Looking on the scheme, DMP can qualitatively estimate a state of a subsystem and define which parameters are outside of a range of allowable meanings (these parameters are allocated by the red color). For more detailed acquainting with a state of the subsystem, the operator has access to the parameter subworkspace.

2. A working space of the urgent messages. At a normal condition of the pump in a working space of the urgent messages there is only one message “System <name> is in norm”. This message has a green background and does not signal about any anomaly. At occurrence of abnormal situations in a working space there are the appropriate messages on a red background.

3. A working space of KB. In it the decision tree is located. Basically this space is intended for an expert and a knowledge engineer for creation and testing KB.

In a decision making mode the means for a choice, concealment, moving and change of the volume of the specified working spaces are given to DMP. With the help of these means he can design project interface with the application.

In the mode of decision making a DMP receives the means for choosing, hiding, transferring and changing a dimension of given working spaces. With the help of these means DMP can configure interface with the application.

Analysis of activity of operative-dispatching personnel has showed that it is preferably to use a three-level system of information representation on a control-labeled object or a process:

- a level of a system (or an object) on the whole, at which to be informed in what a state (normal, abnormal or critical) the system is and in what subsystems the deviations have arisen;
- a subsystem level at which a state of a particular subsystem appears;
- a level of directly measurable parameters with indicating not only parameter values but dynamics of their changing.

For building the graphic images a special editor is used. The basic (first) type of image corresponds to a level of a system (an object) or a process on the whole and has a kind of the sun (kernel) with going out rays corresponding to subsystems (or generalized parameters of a process) SS$_1$ of a top level. The number of rays is defined by a number of subsystems or generalized parameters and a thickness (a size of appropriate labels) of rays is determined by their importance. The kernel colour and rays may be red, yellow or green. The green colour of a kernel corresponds to a normal (on the staff) state (S$_n$) of the whole system (a process), the yellow one – to an abnormal state (S$_a$) in presence of some deviations in subsystems and the red colour corresponds to a critical state (S$_c$). The connection between states (colours) of subsystems and a kernel colour is given by means of production rules storing in KB.

The image of the second type corresponds to a subsystem level and characterizes its state. If the given subsystem SS$_i$ is characterized by a collection of measurable parameters $\{P_i\}$, $i=1,2,\ldots,m$, then the image of the third type occurs corresponding to the parameter level.

Besides the main image types there are means of switching on additional screens mapping dynamics of changing appropriate parameters in the form of graphics and provided by timers. The example of a polyviso for DMP with cognitive graphics is given in Fig. 2.
CONCLUSIONS

During the implementation of the RTSS prototype for monitoring and management of a nuclear station power block, 31 classes of objects, 12 subclasses of variables, 2 subclasses of connections between objects, 7 relations between objects were defined; 38 generalized rules, 45 procedures and 4 functions were written. DB about MCP includes 33 parameters, about CAP – 21 parameters, about EJ – 32 parameters (Eremeev et al. 1999; Eremeev et al. 2003). For each continuous parameter, a subworkspace and objects for graphic display of dynamics of changing its values are defined.

References


Vadim Nick. Vagin is Professor of the Applied Mathematics Department, Moscow Power Engineering Institute (Technical University) (MEPI (TU)). He received a degree of Candidate of Technical Sciences (Ph. D. degree) in Computer Science Department and Doctor of Technical Sciences in Applied Mathematics Department of MEPI (TU). His research interests include inference methods in deduction, abduction, induction; nonmonotonic logics; intelligent decision making systems. He is Academician of Russian Academy of Natural Science and International Informatics Academy. He is Laureate of President Award for the education area.

Alexander Pavlovich Eremeev is Professor of the Applied Mathematics Department, the Head of the Applied Mathematics Department, Moscow Power Engineering Institute (Technical University) (MEPI (TU)). He received a degree of Candidate of Technical Sciences (Ph. D. degree) and Doctor of Technical Sciences in Applied Mathematics Department of MEPI (TU). His research interests include intelligent decision making systems, temporal logics and heuristic search algorithms. He is Academician of Russian Academy of Natural Science and International
Informatization Academy. He is Laureate of President Award for the education area.
SIMULATION OF OVERVOLTAGES FOR SWITCHING OFF LAGGING LOAD FROM MAINS

Jerzy Smyczek and Konrad Zajkowski
Department of Electrical Engineering, Technical University of Koszalin
75 - 620 Koszalin, ul. Raclawicka 15-17,
Poland
E-mail: smyczek@tu.koszalin.pl  konza@tu.koszalin.pl

KEYWORDS
overvoltage, nonlinear, system analysis, transforms, approximation techniques

ABSTRACT

In the paper a nonlinear circuit R.L with nonlinear resistance has been used. The nonlinearity has been assumed as incomplete third degree polynomial. Analysis has been done on the base of approximate transmittance of a system. Overvoltages has been calculated on the ground the MAPLE programme.

INTRODUCTION

Let’s consider the system presented for Figures 1.

\[ e_s(t) = \begin{cases} E_m \sqrt{2} \sin(\omega t + \varphi_{e,0}) & t < 0 \\ E_m \sqrt{2} e^{-\nu t} \sin(\omega t + \varphi_{e,0}) & t \geq 0 \end{cases} \]

\[ R(i) \Rightarrow u_R(t) = ai + bi^3 \]

\[ i(0) = i_0 \]

Figure 1: Examined circuit

Before opening a switching \( W \), a nonlinear resistor \( R(i) \) was connected. Assuming that in \( t < 0 \) there is steady state in the system, since

\[ e_s(t) = \begin{cases} E_m \sqrt{2} \sin(\omega t + \varphi_{e,0}) & t < 0 \\ E_m \sqrt{2} e^{-\nu t} \sin(\omega t + \varphi_{e,0}) & t \geq 0 \end{cases} \]

\[ R(i) \Rightarrow u_R(t) = ai + bi^3 \]

\[ i(0) = i_0 \]

On the base of the second Kirchhoff’s law, we have

\[ L_s \frac{di}{dt} + R_s i + u_R(t) + e_s = 0 , \quad t \geq 0 \]  

\[ L_s \frac{di}{dt} + i(R_s + a) + bi^3 = -e_s , \quad t \geq 0 \]

with condition \( i(0) = i_0 \).

Denoting \( C_1 = \frac{1}{L_s} (R_s + a) \), \( C_3 = \frac{b}{L_s} \), \( z(t) = \frac{e_s}{L_s} \),

\[ K = i(0) = i_0 \]

we get the nonlinear differential equation in a form

\[ \frac{di}{dt} + C_1 i + C_3 i^3 = z(t), \quad t \geq 0 \]

ANALYSIS OF A SYSTEM

In order to work out the equation (7) applying a method presented in (Smyczek 1973, „Determination...”) and (Smyczek 1973, „Using...”) which depend on describing of solution in shape of Voltera series and next applying nondimensional of Laplace transform also the method of connection variables [Appendix 2]. In results, it is given a transmittance \( T_w(s) \). The form the transmittance depends on form of excitation. Assuming a step unit of the \( w(t) = A \cdot 1(t) \), this is \( W(s) = A/s \), and \( T_w(s) \) for equation (7), we get

\[ T_w(s) = -\frac{1}{s + C_1} - \frac{C_3 6 A^2}{(s + C_1)^2 (s + 2C_1) (s + 3C_1) \sqrt{A/V}} \]

While a transform of current – on the base of the appendix is

\[ I(s) = T_w(s) \cdot [Z(s) + K] \]

For the forcing into form (1) after Laplace’s transform, the expression (9) is

\[ [Z(s) + K] = -\frac{E_m \sqrt{2}}{L_s} \frac{(s + \alpha_s) \sin \varphi_{e,0} + \omega \cos \varphi_{e,0}}{(s + \alpha_s)^2 + \omega^2} + K \]

Replacing the values:

\[ E_m = 110,78 \ V , \ \varphi_{e,0} = 5.76 \ rad , \ R_s = 1,00 \ \Omega , \ i_0 = 1,53 \ A , \ L_s = 0,28 \ H , \ a = 1000 \frac{V}{A} , \ b = -400 \frac{V}{A^2} , \ \alpha_s = 1 , \ \omega = 314,1 \ \frac{1}{s} \]

the transform \( I(s) \) has the following form
\[
I(s) = \frac{0.25 \cdot 10^{-15}(100000s^2 + 0.56 \cdot 10^8s - 0.66 \cdot 10^8)}{(100000s + 0.71 \cdot 10^9)(100000s + 0.11 \cdot 10^9)} \cdot \frac{200000s^2 + 0.43 \cdot 10^{10}s^2 + 0.28 \cdot 10^4s + 0.54 \cdot 10^7 + 0.17 \cdot 10^6 \cdot A^2}{25000s^2 + 50000s + 0.25 \cdot 10^9}(200000s + 0.71 \cdot 10^9)^2
\]

The transmittance \( T_w(s) \) (approximated) of a nonlinear system depends on a magnitude of forcing \( A \).

The simulations of current and voltage solution have been realized for different values \( A = 0 \ldots 6000 \). In a range of 0…3000 parameter \( A \) doesn’t influence transient of a current. A solution in time domain of (12), this is
\( i(t) = \mathcal{L}^{-1}\{I(s)\} \), and we get it in form of an analytical formula, but the result is quite complicated, then the final effect is presented in diagrams.

**SIMULATION OF OVERT_VOLTAGES**

A solution of equation (12) has been realized for applying the MAPLE programme. Results of solutions current and voltage \( u_R(t) \) - in accordance with (2) are presented below.

\[
\begin{align*}
\text{Figures 2: A characteristics of nonlinear element} \\
\text{Figures 3: Current at terminal } R(t), a=1000, b=-400 \\
\text{Figures 4: Voltage at terminal } R(t), a=1000, b=-400 \\
\text{Figures 5: Current at terminal } R(t), a=500, b=-100 \\
\text{Figures 6: Voltage at terminal } R(t), a=500, b=-100
\end{align*}
\]

Next, it has been considered a case when
\[
e_j(t) = E_m \sqrt{2}, \quad t > 0
\]

So, the \( Z(s) + K \) get a form
\[ Z(s) + K = \frac{-E_m \sqrt{2}}{L} \frac{1}{s} + i_0 \]  
(14)

and \( I(s) = T_m(s) \cdot [Z(s) + K] \) after replacing the same values, which shown in (11), yields

\[ I(s) = \frac{0.0028125 \cdot 5075s - 0.22 \cdot 10^8}{(703s + 0.25 \cdot 10^7)^2} \cdot \frac{0.22 \cdot 10^6 s^3 + 0.47 \cdot 10^{14} s^2 + 0.31 \cdot 10^{16} s + 0.6 \cdot 10^{21} + 0.19 \cdot 10^{14} A^2}{(703s + 0.75 \cdot 10^7)^2} \cdot s \]  
(15)

The simulations of the \( i(t) \) and \( u_0(t) \) have been done for values \( A=50 \). The solutions are presented in below figures.

Figures 7: Current at terminal \( R(i), a=1000, b=-400, A=50 \)

\[ e_s(t) = E_m \sqrt{2} e^{-at} \sin(\omega t + \varphi_{es}) \]

Figures 8: Voltage at terminal \( R(v), a=1000, b=-400, A=50 \)

\[ e_v(t) = E_m \sqrt{2} \]

The received results have been compared with the previous case when \( e_s(t) \) is in form (1). In both solutions a magnitude of an overvoltage is very similar and equal to 600V in \( t = 0^+ \).

**CONCLUSION**

In the paper an analysis of a nonlinear circuit with nonlinear resistance (varistor) has been presented. This resistor is applied to reduce overvoltages while cutting out an inductive receivers (f.e. electric machines) from mains.

The analysis has been conducted on the base of transmittance of a nonlinear equation, which was calculated in (Smyczek 1973, „Using...“). A simulation has been mode for applying the MAPLE programme.

It has been proved that to optimal a choice of nonlinear element can be assumed a forcing function (extremally) is constant and is equal an amplitude, so an analytical solution is quite simple, and convenient to analysis.

A presented method enables an analysis of the type circuits and in consequently a reducing of overvoltages.

On the base of this method a nonlinear resistor may be selected, and also a change of a parameter of circuit for values of overvoltages may be analyzed.

**APPENDIX 1**

Suppose the equation

\[ L(y)\cdot y(t) = x(t), \text{ with } y(0) = y_0, \quad y'(0) = y_1, \ldots \]  
(1.1)

where \( L(t) \) is a linear differential operator

\[ L(t) = a_n \frac{d^n}{dt^n} + a_{n-1} \frac{d^{n-1}}{dt^{n-1}} + \ldots + a_1 \frac{dt}{dt} + a_0 \]  
(1.2)

and \( y_0 \) -initial conditions.

Assuming the Green’s function \( g(t) \) which is a solving the equation

\[ L(t)\cdot g(t) = \delta(t) \]  
(1.3)

with \( \delta(t) \)-as Dirac’s impulse.

In domain complex frequency \( s \) the Laplace’s transform give

\[ L(s)\cdot T(s) = 1 \rightarrow T(s) = \mathcal{L}[L] \]  
(1.4)

where \( L(s) = \mathcal{L}[L], \quad T(s) = \mathcal{L}[g(t)] \) - is known as operator transmittance.

If replacing as

\[ Z(s) = \mathcal{L}[X] = a_0 y_0 + a_1 y_1 s + \ldots \]  
(1.5)

a solution (1.1) in a complex variable domain it follows that
\[ Y(s) = T(s) \cdot Z(s) \]  

(A.7)

**APPENDIX 2**

Let assume that a differential equation of a nonlinear system have a form

\[ a_n \frac{dy^n}{dt^n} + a_{n-1} \frac{dy^{n-1}}{dt^{n-1}} + \cdots + a_0 y + b y^3 = x(t) \]  

(A.8)

or after dividing by \( a_n \) both the sides

\[ \frac{dy^n}{dt^n} + C_{n-1} \frac{dy^{n-1}}{dt^{n-1}} + \cdots + C_1 \frac{dy}{dt} + C_0 y^3 = z(t) \]  

(A.9)

Next, let \( y(t) \) is an Volterra’s series

\[ y(t) = \int_0^t k_1(t-r) x(r)dr + \int_0^t \int_0^r k_2(t-r_1)k_1(r_1) x(r_2) r_1 dr_1 dr_2 + \int_0^t \int_0^r \int_0^s k_3(t-r_1)k_2(r_1-r_2) k_1(r_2) x(r_3) r_1 r_2 dr_1 dr_2 dr_3 + \cdots \]  

(A.10)

where \( k_1(*) \), \( k_2(*) \), \( k_3(*) \) - Volterra’s kernels

Substituting (A.9) into (A.8) and next doing of the n-th Laplace’s transform we obtain an expression for kernels \( k_n(*) \) and after remaining three term of the series, we have

\[ K_1(s) = \frac{1}{s + C_1} \]  

(A.11)

\[ K_2(s_1, s_2) = 0 \]  

(A.12)

\[ K_3(s_1, s_2, s_3) = \frac{-6}{(s_1 + C_1)(s_2 + C_1)(s_3 + C_1)} \]  

• \( \frac{b}{s_1 + s_2 + s_3 + C_1} \)  

(A.13)

Then response functions are

\[ Y_1(s) = K_1(s) \cdot Z(s) = \frac{A}{s(s + C_1)} \]  

(A.14)

\[ Y_2(s_1, s_2, s_3) = \frac{-6 \cdot b}{s_1 s_2 s_3 (s_1 + C_1)(s_2 + C_1)(s_3 + C_1)} \]  

• \( \frac{A^3}{s_1 + s_2 + s_3 + C_1} \)  

(A.15)

Next, applying the method of the connection variables (Smyczek 1973, „Determination….”), we have

\[ Y_3(s) = \frac{-6 \cdot b \cdot A^3}{s(s + C_1)^2(s + 2C_1)(s + 3C_1)} \]  

(A.16)

Hence, the complete response is

\[ Y(s) = Y_1(s) + Y_3(s) = \frac{A}{s(s + C_1)} + \frac{6 \cdot b \cdot A^3}{s(s + C_1)^2(s + 2C_1)(s + 3C_1)} \]  

(A.17)

Farther because the forcing function \( Z(s) \) have a form \( Z(s) = \frac{A'}{s} \), the transmittance, or transfer function, we obtain finally

\[ T(s) = \frac{Y(s)}{Z(s)} = \frac{1}{s + C_1} \cdot \frac{6 \cdot b \cdot A^2}{(s + C_1)^2(s + 2C_1)(s + 3C_1)} \]  

(A.18)

**REFERENCES**


**BIOGRAPHY**

**JERZY SMYCZEK** was born in Lodz, Poland and obtained his degrees in 1962 (M.Sc.), 1970 (Ph.D.), 1990 (D.Sc.) in Technical Univ. of Lodz (TUL). He worked at TUL in 1962-1990, and 1990-1992 in Technical Univ. of Opole. From 1992 he has been working in Technical Univ. of Koszalin (TUK) as professor electrotechnics and telecommunication. Research domain: signal theory and information, theory of nonlinear systems, systems transmission of information. Head ad Dept.of Electrical Eng. TUK.

**KONRAD ZAJKOWSKI** was born in Gorzow Wielkopolski, Poland. From 1989 to 1992 he worked as an electrician in medical industry. Then he went to the Technical University of Koszalin in Poland where he studied electronics (speciality Automation and Robotics) and obtained his degree in 1998 (M.Sc.). In 1999 he started working for Technical University of Koszalin. He deals with using network in industrial devices and theory of Nonlinear Systems. He carries out research on improvement of methods of overvoltages reduction.
MODELLING OF THE FLASHOVER OF THE POLLUTED ELECTRIC INSULATOR

El Ouni Abdelaziz - Mikhaïl Javoronkov-Ghodbane Fethi - Chaabane Abdelaziz
and Belakhdar Zohra
L.S.E - National School of engineers of Tunis, BP 37 - the Belvedere 1002 - Tunis - Tunisia
E-Mail: abdelazizelouni@yahoo.fr

KEYWORDS
Insulator, pollution, modelling, leakage current, flashover.

ABSTRACT
The phenomena of arcing that happen on a contaminated surface are so much complex that it is necessary to make many simplifying hypothesis and represent them by a mathematical model. The complexity of these phenomena has many factors. We can mention notably the variability of the superficial resistivity, the non uniformity of the insulators form, the distribution of the pollution... In fact, an adequate model can be very useful for the rationalization of practical experiences, for the interpretation and the presentation of test results.

1. INTRODUCTION
The Average and high voltage posts, situated in many places such as the production centre or the consumption centre as well as the connected voltage lines, are submitted to various pollutions. The most frequent pollutions are the marinade pollution, the desert pollution and the industrial pollution. The voltage lines installed to the edge of the sea are exposed to surfs marine carried by the wind, gradually deposit on the insulators and form a layer of salt. This layer, itself humidified by surfs, by a fog or even by condensation, becomes conductive. Next to some desert and industrial zones, the insulators are covered with dusts and sand grains. In conditions of strong humidity, the dissolution of salts contained in these dusts and sand grains lead to the formation of an electrolyte. Indeed, the phenomenon of pollution results from the concomitance of a polluted deposit and the humidification of this deposit. Very often, a site presents several types of combined pollutions. For example the regions of Kebili and Zarzis are known by a marinade and desert pollution.

2. FLASHOVER PHENOMENON
The Pollutants when they are dry, do not pose generally problems, but once humidified by an artificial or natural agent, such as the fog, the morning dew, the condensation, the rain or marine surfs, this deposit becomes conductive. In the presence of the service voltage, a leakage current is established and flowed through the electrolyte that has the effect of increasing both the superficial layer conductivity and the leakage current. The growing heating provokes a local dryness, increasingly large of the polluted layer until the leakage current is elevated in this zone. This leads to a sensitive reduction of the leakage line and the reconciliation of the voltage to the edges of the dry zone. In this case, arcs can appear and develop until provoking the total flashover of the chain insulator [1]. Be that as it may, the description of the principle clarifies the various process, to be considered, in order to achieve a complete modelling of the flashover insulator (see figure 1): thermal process that governs the dryness of the electrolyte and ionisation process that determines the striking and the restricting of the local arcs as well as their elongation until the flashover [2].

3. MODELING OF THE POLLUTION PHENOMENON
Many authors [3 - 4] have tried to reproduce the phenomenon of flashover on simplified models trying to describe it by simple equations. Obenaus has succeeded in establishing the first electrical model in the case of a DC voltage. Concerning the alternative current, the phenomenon of arc is more complex thus it is necessary to proceed to simplifying hypothesis ending to an adequate model.

![Schematic representation of the pollution flashover process](image)
Next, we consider an insulator plan (total length L) on which a partial arc of x length and resistance $R_{arc}$ (as indicated on figure 2) appear. The choice of this representation in two dimensions is justified by the fact that the phenomenon of the flashover depends essentially on the surface characteristics of the polluted layer.

![Fig. 2. Simplified representation of the Arc development](image)

This layer of pollution can be modelled by a resistance $R_p$ in parallel with a capacity $C$. Hence, the equivalent model is shown in the electrical diagram of figure 3.

![Fig. 3. Equivalent Electrical diagram of the polluted insulator](image)

With $i_{arc}$: arc current, $i_b$ and $i_c$ are the two active and capacitive components representing the current in the polluted layer.

For the plan case, $R_p$ is expressed in the next manner:

$$R_p = \rho_p \frac{(L-x)}{S} \quad (1)$$

The capacity corresponding to the pollution layer $C$ is calculated by making the linear approximation [5]:

$$C = \epsilon \frac{S_p}{(L-x)} \quad (2)$$

If we consider that the arc is a canal of cylindrical section $S$, length $x$ and resistivity $\rho_{arc}$, its arc resistance will be:

$$R_{arc} = \rho_{arc} \frac{x}{S} \quad (3)$$

For the case of a sinusoidal voltage, the expression of the admittance of the circuit on figure 3, is as follows:

$$\frac{Y_e}{R_{arc}} = \frac{1}{R_p} \quad (4)$$

By replacing $R_{arc}$ and $R_p$ by their expressions, the admittance equation becomes:

$$\frac{Y_e}{S_p (1 + \sigma^2 \rho_p^2 \varepsilon^2)} = \frac{S_p (1 + \sigma^2 \rho_p^2 \varepsilon^2) + S_p \rho_p (L-x) (1 - j \sigma \rho_p \varepsilon)}{S_p (1 + \sigma^2 \rho_p^2 \varepsilon^2)} \quad (5)$$

4. VOLTAGE EXPRESSIONS

4.1. Arc voltage

To determine the expression of the arc voltage, we must consider the critical case of the criterion of flashover propagation [6].

$$\frac{S_p \cdot \alpha \cdot \rho_{arc}}{\beta} (S \cdot \rho_p - S_p \cdot \alpha \cdot \rho_{arc}) = 0 \quad (6)$$

With:

$$\alpha = 1 + \sigma^2 \rho_p^2 \varepsilon^2$$

$$\beta = (\rho_{arc} \cdot S_p \cdot \alpha - \rho_p) S^2 + \sigma^2 \rho_p^4 \varepsilon^4 S^2$$

The arc voltage can be expressed by:

$$V_{arc} = \frac{I_{arc}}{Y_{arc}} \quad \text{with} \quad Y_{arc} = \frac{S}{\rho_{arc} \cdot x}$$

In fact, the energy provided by seconds and by the unit of length to the electrons is expressed by the next equation [7]:

$$P = \frac{E_{arc} \cdot I_{arc}}{l_{arc}} \quad (7)$$

With:

$$E_{arc} = \frac{N}{l_{arc}}, \quad V_{arc} = x \cdot E_{arc} = \frac{x \cdot N}{l_{arc}} \quad (8)$$

Or again:

$$V_{arc} = \frac{x \cdot N}{l_{arc}}, \quad \text{with} \quad n = 1$$

N and n: Parameters depend on the milieu in which the discharge is burnt. The expression of the analytically obtained arc voltage is similar to those established from the empirical formulae [3 - 4 - 8].

4.2. Voltage of supply

The expression of the circuit voltage is given by the next relationship:

$$U = V_{arc} + \frac{I_{arc}}{Y_p} \quad (9)$$

If the pollution layer is resistive: $Y_p = \frac{1}{R_p}$; If we replace $V_{arc}$ and $R_p = \rho_p (L-x)$ by their expressions, we will have:

$$U = \frac{x \cdot N}{l_{arc}} + \rho_p (L-x) \cdot I_{arc} \quad (9)$$
5. FLASHOVER CONDITIONS

5.1. Critical Current

If we resume the expression of the voltage and derive this equation as compared to x, we will obtain:

\[ I_c = \left( \frac{N}{I_p} \right)^{1/n} \]  \hspace{1cm} (10)

5.2. Flashover Critical Voltage

To find the equation of the flashover voltage we must replace the expressions of critical current and critical distance in the voltage expression:

\[ U_c = \frac{N}{n+1} \left( \frac{L}{n+1} \right) \frac{1}{r_p} \left( \frac{N}{r_p} \right)^{n+1} \]  \hspace{1cm} (11)

By developing this expression, we will have:

\[ U_c = L \cdot N^{n+1} \cdot r_p^{n+1} \]  \hspace{1cm} (12)

In the case of a real cylindrical insulator, the lineique resistance can be expressed, with respect to the surface conductivity, by the next relationship [9]:

\[ r_p = \frac{1}{\pi D_{eff} \cdot \sigma} \]  \hspace{1cm} (13)

with: \( D_{eff} = \frac{1}{\pi f} \); \( D_{eff} \): The effective diameter insulator; \( f \): Factor of form. By taking into account these parameters, the flashover critical voltage becomes:

\[ U_c = (L \cdot N)^{n+1} \cdot \left( \frac{1}{\sigma} \right)^{n+1} \]  \hspace{1cm} (14)

6. MODEL VALIDATION

To validate our model, we are going to consider models brought in the literature, especially that of Claverie, Porcheron [10] and Obenaus [3]. By comparing the characteristics \( U_c = f(s) \) of the obtained data, by these different empirical modelling, with our analytic model (figure 4), we notice a good concordance.

![Fig. 4. Flashover voltage with respect to the surface conductivity](image)

- Experimental validation of the model

The results of these tests on the standard insulator of the two sites of Kebili and Zarzis are given on figures 6 and 7.

![Fig. 5. Critical voltage with respect to the saltiness degree of the two sites](image)

![Fig. 6. Current according to the voltage for a S = 7.6 S/m (site of Zarzis)](image)

The results presented above show very well the model validity with a good concordance between the practical and theoretical results.

7. CONCLUSION

In this article, we have developed an analytic model that allows us to determine the flashover critical voltage. This is based on the energy balance, the equivalent electrical circuit and the physical characteristics of the arc. To take into account the geometry of the real insulator, we have introduced a factor form in the analytic model. The results given by the elaborated model converge with those obtained by the known empirical models as well as the practical tests realized in the tests and measure center of the STEG (Tunis).

8. BIBLIOGRAPHY

TRAFFIC AND LOGISTICS
TRAFFIC SIMULATION
APPLICATION OF REFLECTIVE MODELS OF A TRANSITIC SYSTEM

Thierry Coudert
LESTER, Université de Bretagne Sud
Centre de recherche
Rue St. Maude, BP 92116
F-56321 Lorient CEDEX, France
E-mail: thierry.coudert@univ-ubs.fr

Pascal Berruet
LESTER, Université de Bretagne Sud
Centre de recherche
Rue St. Maude, BP 92116
F-56321 Lorient CEDEX, France
E-mail: pascal.berruet@univ-ubs.fr

Eugene Kindler
Ostrava University Faculty of Sciences
CZ – 701 03 Ostrava, Dvorakova 7, Czech Republic
E-mail: kindler@ksi.mff.cuni.cz

KEYWORDS
Anticipation, Nested simulation, Transitic Systems, Piloting

ABSTRACT
The aim of this paper is to present a new simulation approach based on nested simulation in order to anticipate the behavior of the system. The authors focus in this paper on Transitic systems. These systems have many flexibilities and have to be on-line piloted. The issue considered in the present paper is the assignment of a parcel to the system. The approach is tested on an industrial conveyor with rollers.

1 INTRODUCTION
The manufacturing systems of the 21st century constitute a complicated mix of humans, software systems, processes and equipments. These systems have to support rapid changes in the economic, technical and organizational manufacturing environment. They can be considered as series of physically networked processes. They can be performed manually or with automated equipment. The main characteristics of manufacturing systems are: enterprise integration, distributed architecture, heterogeneous environments, integration of humans, co-operation, open and dynamic structure and fault tolerance (Shen and Norrie 1999).

This paper considers the simulation of a special class of manufacturing systems: the Transitic Systems (TS). The principle is based on the nested simulation that provides a real benefit for the decision making procedure. In section 2, the transitic systems are defined. Section 3 presents the principles of component-based simulation of TS. Section 4 presents an example of a TS, named IUP conveyor. Section 5 presents questions about the control of IUP, which lead to simulation during the real time control. In Section 6 information is presented on a result of synthesis between simulation in real time control and simulation during a system design; the result is nested and reflective simulation. The obstacles carried by these kinds of simulation are presented in Section 7. Section 8 presents the reflective simulation of IUP. In Section 9, conclusions are formulated.

2 DEFINITION OF TRANSITIC SYSTEMS
Transitic systems (TS) are particular manufacturing systems that transport products (or parcels) from different locations with a fast flow and dispatch them to their right locations. They are composed of different types of conveyors, elevators, consignments, sorters, and automated guided vehicles. Conveyors can be linear, curved, and circular. They can have pneumatic jacks, stops and switchings.

Different features characterize transitic systems:
(1) They are specific manufacturing systems, in the sense that no transforming resources are considered. Only storage and transfer resources are taken into account.
(2) They are Discrete Event Systems (DES). Their control is mainly discrete.
(3) They also include a continuous dimension. Because of the speed transfer, such systems are often analyzed in terms of flows, capacities and costs.
(4) Few types of elements are used to design these systems. The reusability concept is very important, enabling designers to reuse some validated parts.
(5) They include flexibility notion. Different paths enable the same area to be reached. The management of the traffic of such systems has to guarantee good performances avoiding deadlock and collisions (Fanti 2002) reducing work-in-progress and minimizing products tardiness.

3 COMPONENT BASED APPROACH
TS can be simulated, either during their design or during their real time control. To implement their simulation models, property (4) favors the component-based approach: in such a
case, writing a simulation model corresponds to composing the modeled system of its elements. Object-oriented programming is a good programming paradigm for that purpose. Although there are a lot of object-oriented programming languages at present time, SIMULA (Dahl et al. 1968; Simula Standard 1989) is used. The reason is that this language is not only object-oriented but also process-oriented and block-oriented. The profit of the presence of these three orientations is presented in section 6.

In SIMULA have been declared classes like parcel, place, segment, junction, branching, confluence, motor, barrier, jack, etc., and facilitated them by methods like move_step, return_step, move_through(segment), move_to(place), switch, set_to(segment), return_to(place), move_to_end_of(segment), continue_to(direction, length), branch_to(direction1, length1, direction2, length2), branch_into(segment1, segment2), accept(segment), connect_from(segment), etc. Note that segment represents a straight part of a TS that contains no branching and no confluence at its internal places; such elements can occur at its beginning place and at its end place. It has its length and orientation and is composed of instances of class place. X.connect_from(Y) closes a cycle composed of segments so that a new segment is generated leading from the end of Y to the beginning of X. When two segments are generated (e.g. in case of branching), they automatically get names last_segment and last_but_one_segment. When one segment is generated it automatically obtains name last_segment and the segment pointed originally by last_segment receives name last_but_one_segment. Also tools for directions of the segments were prepared in a form of classes, their instances like up, down, right and left, and a method sum for their angular addition. There are 27 such classes and 46 such procedures, which can be applied by any user who wants constructing a simulation model. They represent a result of the corresponding domain analysis (Hill 1996), i.e. the analysis of the domain of the transitics systems; so they form a simulation programming language oriented to this domain. The classes are described in some more details in (Kindler et al. 2003).

4 AN EXAMPLE – CONVEYOR IUP

Let us illustrate these concepts on an example. Its static structure can be seen in Fig. 1. It is a conveyor called IUP according to the acronym of the title Institut Universitaire Professionnalisé (a component of University of South Brittany in Lorient, France) where it is settled and studied. It consists of a great cycle called main ring and of five working areas. Parcels enter the main ring at the beginning of its lower long component and are transported to working places according to their technological programs. The transport proceeds according to the arrows. Both the long components of the main ring carry the parcels by motorized rollers while all other components of the system are deserved by pneumatic jacks. Parcels can be stopped due to mechanical stops. The main component of every working area is the short segment, parallel with the long segments of the main ring. It is called stopper. A parcel can access subsequently three places A, B and C (at Fig. 1, they are mentioned at working areas W1A1 and W4A3). The parcel is manufactured at B, but it can enter a working area and wait at its place A, if B is occupied. When a parcel has been manufactured it comes to place C and then can leave the working area for the main ring. In case its injection on the main ring could cause a crash with another parcel there, it has to wait at C. If a parcel is supposed to enter a working area W but the place A is already occupied, it cannot enter W. It has to go on along the main ring and look for another equivalent working area. According to its technological program, such a working area can be different from W or not. In the last case the parcel has to go on along the whole main ring. The elaborated parcels leave the conveyor at the same place where they entered.

![Figure 1: Static structure of the conveyor IUP](image-url)
Using the language mentioned above, a class \( WP \) was formulated, representing the common properties of the working areas. The intention is to make the description of the system as short as possible, and therefore \( WP \) covers not only what we described above as working area, but also so called \( chord \), i.e. the short segment of the main ring, which is parallel to the stopper, and \( exit \), i.e. the next continuation of the chord at the main ring. So the basic structure of the described system is decomposed into five instances of \( WP \), two shorter segments of the main ring (represented as vertical ones in Fig. 1) and two start segments of the long components of the main ring:

```plaintext
class WP(entry, continuing);
    ref(segment) entry;
    short integer continuing;
begin ref(segment) stopper, chord, exit;
    ref(place) A, B, C;
    ref(direction) procedure turned(x);
    ref(direction) x;
    turned:=sum(x, entry.orientation);
    entry.branch_to
        {turned(up), 7, turned(right), 3};
    chord:=last but one_segment;
    last_segment.continue_to
        {turned(up), 7};
    stopper:=last_segment;
    A:=stopper.first_place;
    B:=A.suc; C:=B.suc
    chord.continue_to
        {turned(up), continuing};
    exit:=last_segment;
    exit.connect_from(stopper)
end;
```

procedure \( turned(x) \) returns the direction \( x \) modified according to the orientation of the entry of the working area. The description of the basic structure of the whole system is as follows:

```plaintext
arrival:=new segement(30,14,right,10);
WP(1):=new workplace(last_segment,20);
WP(2):=new workplace(WP(1).exit,14);
WP(2).exit.continue_to(up,9);
last_segment.continue_to(left,8);
WP(3):=new workplace(last_segment,13);
WP(4):=new workplace(WP(3).exit,13);
WP(5):=new workplace(WP(4).exit,4);
arrival.connect_from(WP(5).exit);
```

Of course, the description of the basic structure has to be completed by setting a motor and pneumatic jacks at the given segments and mechanical stops at the given places. One common motor called \( m1 \) is generated and connected with all parts of the long components of the main ring by statements \( X.movement=m1 \), where \( X \) means the name of the part. So the formulation of class \( WP \) has to be completed by a statement \( chord.movement=m1 \) and similarly for \( arrival \) etc. The jacks are set to segments similarly – for example the stopper of every instance of \( WP \) is facilitated by its proper jack by statement \( stopper.movement=-new jack(...) \) occurring in the declaration of class \( WP \) (in brackets, parameters of the jack occur). The mechanical stops are set to particular places as e.g. \( chord.first_place.suc.blocking=new barrier \) (read “the pointer blocking of the successor of the first place of segment \( chord \) is a new barrier” – in case a place has no mechanical stop its pointer blocking has value \( none \), which is the default value for any place).

The last phase of the description of the given system consists in rules concerning the movement of parcels in the system. The rules are not simple but they use procedures like \( move..., switch \) etc., usual algorithmic tools (conditional statements, cycles etc.) and common scheduling statements of SIMULA (e.g. \( hold(t) \), meaning that running the sequence of the rules must be interrupted until the (simulated) time grows up of \( t \)).

It is easy to imagine several instances of IUP-like conveyors, which differ in their parameters (motors etc.). Moreover, such instances can function in different environment, i.e. under different rules for the input of entering parcels and for their processing at the working areas. Note that the transitic systems do not reflect details of the processing but only several relevant parameters. At least the duration process is important for the interface of instances with their environment. So the IUP-like conveyors can be studied under their operating under different statistical rules, according to which the parcels come and are manufactured. A frequent task is to balance the resources of the designed IUP with such rules in order the designed conveyor to be optimal. It can be solved by means of simulation; using it, the designer can anticipate the behavior of the conveyors with various parameters of various demands coming to the conveyor from its environment. Because of it, the prepared model was endowed with input data that can model the rules of the environment demands.

Such a model has a certain value when a conveyor is being designed, i.e. when the conveyor does not physically exist but is proposed to exist, namely with certain parameters. Let the model be called \( external model \) and the simulation \( external simulation \).

5 QUESTIONS ARISING DURING THE OPERATING OF IUP

When the conveyor has been designed and realized as a physically existing system, questions may arise. Some of them cannot be answered but with simulation support. They are often questions related to an anticipation of some future development of the conveyor. A model used for that sort of simulation exists in parallel with the existing IUP. Let it be called \( internal model \) and its application \( internal simulation \). Let us present some questions for answering of which the internal simulation is useful.
(a) A fault occurs at a working area that can be supplied by other working areas. The repair would cause an interruption of the whole moving at the conveyor. The question is about the developing of the queue of the parcels “in front of the conveyor” (i.e. of the parcels that request to be served by the conveyor in order to be transported to working places); simulation of the conveyor operating with four working areas can answer the question and therefore help in the decision whether the fault should be repaired immediately or after some time (when e.g. a series of certain parcels will have been performed).

(b) Only four working areas belonging to a set $W$ are used. The fifth area $Q$ is currently stopped due to the maintenance of a processing element. A fault occurs to a working area $P$ belonging to $W$. The question is whether $Q$ can supply $P$, or – better – how long the conveyor will be able to work in a good manner in case $Q$ supplies $P$. That can be told by simulation.

(c) Many parcels move on the conveyor. A new parcel approaches to it and requests to be served by it. The question is whether the parcel should be placed at the input place of the conveyor immediately or whether it would be suitable to let it wait outside the conveyor during some time. If a parcel is placed to the conveyor, it could perform a whole cycle at the main ring and in such a case there would be better to let it wait for some time, because at the main ring it could block movement of other parcels (e.g. of those that return from a working area to the main ring). Simulation can tell whether the parcel, virtually set on the conveyor, can get a place at a working area.

(d) A parcel $P$ approaches the conveyor and a problem like in (c) arise, but more is requested: suppose the working area for handling $P$ can be chosen from a set of several working areas. What choice has the best consequences? One technique is that the destination of the parcel is the working area with free place $A$ or $B$. Nevertheless, when only $A$ is free, it is not excluded that $P$ will wait at this place a rather long time, contrary to the fact that when another working area is more distant but with free $B$, the parcel will be elaborated sooner. But if the priority destination of the parcel will be determined relating to the working areas with free place $B$ it may happen that some other parcel enters the destination area of $P$ sooner than $P$. (simulation demonstrated us that such phenomenon can be repeated, $P$ was presented to cycle rather infinitely). Nevertheless, simulation can show what can be expected when $P$ will be reaching its destination area and so one can simulate the future processes influenced by any logically possible choice of the destination; so one can choose the optimal destination area.

(e) After having been processed in a technological step, a parcel leaves a working area where the step was performed, but the technological program for a parcel requests another technological step that cannot be performed at the same working area. To determine the working area, which would imply the optimal consequences, is a similar question as that presented in (d). Therefore the main technique of internal simulation offers is being applied: moves to all working areas that can be in principle new destinations for the parcel are simulated (possibly the processing of the parcel is added to the simulation process) and the destination for that simulation that demonstrated the best future is accepted.

(f) Questions can also concern the ordering of technological steps: some of them have to be performed in a given order that cannot be changed, but for some groups of products enables flexibility in the order of their technological steps; naturally, the optimal order has to minimize the whole time used for performing the whole group of the technological steps. The decision depends on many aspects concerning the instantaneous state of the conveyor and some future states as well. Simulation can anticipate the future states and their mutual interaction with the technological steps of the group.

The presented questions can be combined to give a more complex request.

There is no essential problem to make simulation models mentioned in the paragraphs (a) – (f) and to include them into the control program of a given conveyor. The only constraint is that the simulation experiments with such models have to be very fast. It can be satisfied in case the simulation models themselves do not use interrogative scheduling statements (i.e. the statements of a form “interrupt the running algorithm until a given condition is satisfied” (Dahl 1966; Genuys 1968)) but only imperative scheduling statements (i.e. the statements of a form “interrupt the running algorithm until the simulated time accesses a certain value” or “interrupt the running algorithm until a signal from another algorithm comes” – see loc. cit.). The simulation programming tools are constructed so that the simulation models compiled from them prefer one type of the scheduling statements; some of them do it for interrogative scheduling statements (e.g. GRAFCET, GPSS), the other do it for imperative scheduling statements. Although using interrogative scheduling statements seems to be more convenient for describing complex systems the demands for the computing speed of the compiled model speak in favor of the simulation tools, which lead to imperative scheduling statements. We used SIMULA, because its standard simulation tools satisfy that demand.

6 NESTED AND REFLECTIVE SIMULATION

At the phase of design of a system, external simulation models are used. If the designers know that the system will use a control computer that will apply simulation (i.e. that will handle with one or more internal simulation models), that computer (including the internal models) has to be reflected by the internal model. Namely, the computer itself does not need to be reflected in its many details, but the run of the in-
ternal models at it must be reflected in details. This statement was proved in (Kindler 2000a) and the solving of programming difficulties was described in (Kindler et al. 2003) and (Kindler 2000b). In the same papers, it was also shown that the commonly used simulation languages, packages (in object-oriented languages, too) and programs are implemented so that they enable to model only one time axis (and therefore they cannot help to implement such nested simulation models). And in the same papers the true help in the nested simulation was also shown, which is presented by general purpose programming languages that are object-oriented, block-oriented and process-oriented (the last property can be formulated also as agent-oriented, as processes combined with object orientation can be simply converted to agents (Kindler 2000c)). Using such a language, a formulation of a simulation experiment (i.e. a description of the simulated system) can be a subblock, which can be nested in any block, therefore inside another description of a simulated system. Namely the description of the internal simulation model can be nested inside the description of the external simulation model. Let us introduce the terms external simulation block and internal simulation block for these blocks. Also from this view point SIMULA appeared a very suitable programming tool.

A special case of nested simulation is reflective simulation, in which the internal model is formally similar to the external model, because it models the same object as the external model. It allows using the same classes introduced by the object-oriented paradigm. They can be imported into both the external simulation block and the internal one. It enables an advantage that both the external and internal blocks can be formulated in the same "language". Especially in using SIMULA, the computer memory is so spared that both the imports use the common memory. The object orientation of the applied "language" enables to include simulation tools into the imported language, i.e. import these tools into both the blocks, among them simulated time axis, too. Therefore each of such blocks has its own time axis and when the computing occurs inside the internal simulation block, both the time axes exist. The most difficult programming work is to prepare the part of the external model reflecting the activity of the modeled simulationist, which he performs when observing its environment and constructing the internal model. In other words, this difficult part concerns the copying of the instantaneous state of the external model and the interpreting it as the initial state of the internal one. The difficulties consist in two facts:

(i) both the models are described in the same language and one could make an error of transplantation consisting in mixing elements of both the models – e.g. including an element of a model into a queue belonging to the other model;

(ii) SIMULA is preserved against transplantation by certain strong syntactical limitations and it is a known phenomenon (Kindler 2000b) that during a long time one thought that it did not allow any communication between the external and internal models, and that it was not sooner than after 25 years of its existence when a leap-frog of that restriction was discovered (without violation of the security against transplantation).

7 CONCLUSION - REFLECTIVE SIMULATION OF THE CONVEYOR IUP

When a TS is being designed, its optimal configuration and optimal control rules use to be tested by simulation. In such a case, all points (a) to (f) presented in section 5 necessarily lead to reflective simulation. Models reflecting aspects (a) and (c) were implemented and run. At the present time, the authors prepare models corresponding to (b) and (d) (Berruet et al. 2003).

References


Kindler, E. 2000c “Simulation of systems with imaging components”. In: Workshop 2000 – Agent-Based Simulation (Maiz 2-4, Passau, Germany). SCS European Publishing House, Ghent, 239-244.

AN INDIVIDUAL-BASED MODEL FOR TRAFFIC SIMULATION IN VIRTUAL ENVIRONMENTS

Tao Ruan Wan and W. Tang*
School of Informatics, University of Bradford,
Great Horton Road, Bradford, BD7 1DP, United Kingdom
E-mail: T.Wan@bradford.ac.uk
*School of Computing, University of Teesside
Brough Road, Middlesbrough, TS1 3BA, United Kingdom
E-mail: W.Tang@tees.ac.uk

KEYWORDS
Autonomous virtual agents, individual-based vehicle model, traffic simulation, 3D virtual environment

ABSTRACT

This paper addresses a microscopic simulation of roadway traffic in 3D virtual environment. An approach of using individual-based vehicle models is proposed, which captures the driving characteristics of each vehicle on the road, such as vehicle's dynamics, path selection and driving behaviours or attitudes. The 3D virtual environment presented in the paper is capable of constructing various roadway network configurations and simulating the different traffic situations. A traffic flow in the system is modelled as a large flow of autonomous vehicle agents, each is simulated individually according its motion behaviour characteristics and the traffic flow is the result of collective behaviours of the vehicle agents. The system models the design layout and the conditions of roadways, traffic control signals, as well as the driving behaviours of all the individual vehicles. A number of simulation tests are also discussed in this paper.

INTRODUCTION

With the rapid development of computer power in recent years, computer simulated traffic systems have become an effective means of assessing the traffic conditions for various design alternatives of transportation facilities, which has emerged as an important evaluation tool for intelligent transportation system (ITS) strategies in recent years (Navin 1986). In the context of traffic simulation, numerical results can provide useful and detailed quantitative descriptions of what could occur, while visual simulation can provide insights into the overall performance and consequences of a specific traffic flow situation, from which a trained observer can easily gain an understanding about the behaviour of the vehicle-road network systems and diagnose what could go wrong or be improved for better traffic performance (Owen et al. 2000). Furthermore, 3D virtual environments add a new dimension to traffic simulation, which can realistically represent and describe the behaviour of roadway traffic flows and provide rich visual information for traffic assessments (Tang and Wan 2002). Three-dimensional virtual reality systems are also useful tools for training and practicing in traffic control. For example, in addition to assessing road designs and monitoring traffic, the system simulations can be used for training personnel in emergency situations. However, there is relatively less work reported in this area compared with the conventional two dimensional traffic simulation systems. As the simulation systems deal with large numbers of social agents (cars, pedestrians, motorcycles, and bicycles) and each of them has its own behaviour characteristics, the challenge is to solve the problem of modelling the massively distributed parallel behaviours (Reeves 1983). However, individual characteristics of traffic elements including human factors, vehicle specifications and dynamics, could be very difficult to model using traditional traffic flow theories or macroscopic simulation approaches (Gipps and Marksjo 1985).

In this paper, we present an individualised intelligent vehicle model for traffic simulations. The three dimensional virtual reality simulation system developed based on this approach is capable of modelling large volumes of traffic in the large complex urban environments and the system is capable of simulating different traffic conditions with various traffic control managements and operations. The simulation model described in the following sections of the paper is based on modelling the motion behaviours of each individual virtual vehicle agent using a set of object-orientated behavioural simulation components, which are acting upon each virtual agent and managing its behaviours according to the external condition statements at each simulation time frame.

TRAFFIC FLOW THEORY

The traditional traffic flow theory studies the relationship between three fundamental traffic variables, in terms of traffic flow velocity \( v \), the density \( p \) and volume or traffic flow \( q \), which could be expressed as \( \dot{q} = vp \). Those relations could be static or dynamic or time dependent relationships. The dynamic models could be derived from the Navier-Stokes equations and the models are referred as the fluid dynamic models. The theory has been used in traffic simulation for a quite long period of time and
received considerable attentions in transport research community. More importantly the theory gives systematic insight into traffic dynamics in some cases (Gazis 1974). However, the model has some shortcomings, such as that there has never been an agreement on the function for speed and density relations, which is required for the traffic model (Persaud and Hall 1989), which could produce unrealistic traffic behaviours, such as backward moving vehicles in a traffic flow (Daganzo 1995). In addition, the traffic flow theory lacks the power of incorporating the individual characteristics of a traffic element. On the other hand, microscopic models, such as car-following model can produce realistic simulation results, in which behaviours of each vehicle are modelled with respect to the vehicle ahead. The advantage of car-following model is that it takes into account of the static traffic flow density as well as the car following characteristics. However, most microscopic traffic simulation approaches including car-following models are computationally expensive, especially for large-scale traffic simulations (Junchaya and Chang 1993). Recently the approaches using particle models for traffic flow simulations have been reported (Russell et al. 1994). In such approaches, a roadway network is represented as a string of cells, and vehicles in the network are modelled as moving particles. Different to the car-following model, vehicles in the particle model move forward in a hopping manner in order to occupy the cell ahead that is available. Therefore, both the road network configuration and vehicles’ movement are discrete. The advantage of the discrete approaches is the improvement on computational efficiency. The particle model approaches lie between that of fluid dynamics and car-following models.

In this paper we present a new approach, which is different from the methods discussed above. In our approach, a roadway network is represented as strings of cells and vehicles in the roadway-network are modelled as a continuous flow. The discrete space for the roadway network provides a framework for high-speed system update and traffic management registration, while a continuous space and time for modelling vehicle dynamic movements produces realistic traffic flow. Therefore the approach takes the advantage of discrete models at the same time maintains the features of fluid dynamic models. The model is capable of taking into account the human factors for more accurate simulation results. Compared with existing systems, which are mainly two-dimensional simple graphical representations, a three-dimensional visualisation system is developed based on the approach. At the current development stage, our system is able to simulate small and medium scale traffic flows of urban environments.

THE INDIVIDUAL-BASED VEHICLE MODEL

Nowadays, especially in developed countries, most traffic-road projects are of the extensions to the existing traffic systems and the scales on them are usually not too large, and it is more so in densely populated areas. Any modification or extension to the existing system must be carefully evaluated in order to gain the maximum performance of a road network. Our individual-based simulation model can be scaled up to simulate medium scale traffic flow environments at the current development stage.

Configuration of A Roadway Network

In our approach, the road-network is modelled as a cell-based system and an individual intelligent model was developed for vehicles so that the model is able to incorporate more individual characteristics of the vehicles. Each roadway configuration maintains a table of these cells. As shown in Figure 1, a vehicle agent is an individual element, which travels from one location to another via a route with dynamic velocity $V$ through a list of cells along the roadway network to destination.

![Figure 1: Description of a roadway network system](image)

Traffic congestion and uncertainty will occur when traffic demand is higher than supply, which brings inefficiency to all road users. Based on the theoretical relationship between vehicles, density, volume, and speed, a simulation system would be used to assist and evaluate traffic flow characteristics. During the simulation, the state of each cell is updated according to the traffic flow. A cell is registered as occupied when a vehicle gets into it. Otherwise it is registered as an empty cell. There will be no difference for the vehicle’s exact poison to the road situation as long as the vehicle is in the area defined by the cell configuration at the time of interest of the simulation. The network configuration for a particular urban environment is loaded from the configuration map of the network at the initialisation stage.

Perception

Research has shown that the perception of human vision plays an important role in human behaviours. Therefore the vision information can add more advanced behaviour descriptions (Renault 1990). The eyes actively perform problem-oriented selections and process the information from the visible world under the control of visual attention (Rybik et al. 1998). Because the simulation of the individual behaviour of vehicles in a 3D virtual environment are only approximations of real-life situations to a certain extent. In order to decompose the complexity of the scene, assumptions must be made on the weather and the visibility condition of the environment. Each vehicle in the system has been assigned a circular vision zone representing the mirrors and the view of the front screen. As shown in Figure 2, a sensor is attached to a vehicle for
its front view with dynamic changeable field of view according to the traffic situation. Approaching a roundabout requires a larger view than that for driving on a straight road.

![Figure 2: Description of the perception of virtual vehicle agents; Vi – velocity of vehicle i, Ri-perception radius of i, Vj- velocity of vehicle j, Rj-perception radius of j.](image)

**Individual-based Vehicle Dynamic Model**

This section discusses the task of tactical driving. There are a number of factors which must be taken into account in terms of human factors, driving force, vehicle-following and rule based vehicle control strategies. Basically, the core of a driver-vehicle system is regarded as a person-machine control system. A molecular model of traffic flow is developed in which an individual human-vehicle has been recognised as comprising a unit of analysis. Important human performance characteristics can therefore be identified to aid in the formulation although no clear solution has yet been found. The individually based vehicle model proposed for simulating the traffic flow uses discrete control strategies. The human-vehicle system has to be discretised into basic components of driving performance including, perception-reaction time, various response times (responses to movements of other vehicles, road conditions, traffic control signals etc.), handling of hazards in the road and individual driving attitudes and other related factors. The second group of factors that must be considered include steering angles, braking force, and speed control etc., which must be described in the model. Next, lane keeping, vehicle following, gap acceptance and stopping. View sensors must also be considered in developing the model. Each individual vehicle must follow its own route whilst keeping a safe distance away from the immediate neighbouring vehicle ahead of it, which is dependable on the vehicles’ speed.

The driving task for the individual models can be divided into a number of subtasks: The visual information received will be analysed, which will identify the other vehicles nearby, the distances between the vehicle and other vehicles, the speeds and the road conditions. The interpretation of the visual information will be carried out in order to provide a basis for decision-making. The AI algorithm will make decisions for updating control inputs to each vehicle. A knowledge database is constructed which encompasses the vehicle characteristics including the driver’s past driving experience. The database consists a set of rules, such as accelerating, braking, and taking a turn.

The general descriptions about a driver’s driving attitude are also included into the system. The control task is concerned with executing the control commands with smoothness and coordinating with timing and space.

**Path Planning**

The average travel time for the vehicle passing a specific route can be calculated. The vehicle will choose the least cost route to travel. The cost function is related to the specific route based on the origin-destination matrix, traffic flow density, and road conditions. A* algorithm for path planning is implemented in the system. In an event of traffic congestion, the vehicles on the same road will make decision to select a different route to reach its destination, therefore. A new path is calculated with respect to the vehicle’s current location.

**The Control of Vehicles**

The internal state structure of each individual vehicle in the system has been designed as a finite state machine (FSM) in which the state of a vehicle changes from one state to another based on its current state and the input it receives, as depicted in Figure 3. At each simulation frame, the state of the vehicle has a specific set of outputs. In general, a driver pays most attention to the changes of driving signals from the vehicles that are directly next to his/her vehicle in all directions. Therefore, a set of rules has been defined to implement the self-contained individual properties of the vehicle.

![Figure 3: simulation system architecture](image)

A FMS is a set of condition and action statements. The transition between the states of a vehicle is performed by the behaviour modelling functions. An example of a FSM of an individual vehicle is implemented as table 1. The vehicle model developed has three degrees of freedom, which can be represented as a set of motion parameters in terms of mass, accelerations and steering angles as well as external force conditions, such as air resistance and ground friction. The dynamics of an individual moving vehicle...
must follow the basic laws of motion, which may be represented as a set of general ordinary deferential equations. Traffic flow is the collection of the individual behaviours of vehicles on the road. A behaviour modelling system updates the internal state of the individual vehicle consistently.

Table 1: FSM statements

<table>
<thead>
<tr>
<th>Condition 1: go forward</th>
</tr>
</thead>
<tbody>
<tr>
<td>If in the vision fields (no other vehicles and no roundabout)</td>
</tr>
<tr>
<td>Action: go forward in a cruising speed</td>
</tr>
<tr>
<td>Action: go forward slowing down</td>
</tr>
<tr>
<td>Action: go forward speeding up</td>
</tr>
<tr>
<td>Else:</td>
</tr>
<tr>
<td>Action: stop and wait until the road is clear</td>
</tr>
<tr>
<td>Action: slow down to keep a safe driving gap</td>
</tr>
<tr>
<td>Condition 2: Turn right</td>
</tr>
<tr>
<td>If in the vision field (no obstacles/other vehicles)</td>
</tr>
<tr>
<td>Action: turn right and go forward</td>
</tr>
<tr>
<td>Else:</td>
</tr>
<tr>
<td>Action: stop and wait until road is clear</td>
</tr>
<tr>
<td>Condition 3: turn left</td>
</tr>
<tr>
<td>If in the vision fields (no other vehicles)</td>
</tr>
<tr>
<td>Action: turn right and go forward</td>
</tr>
<tr>
<td>Else:</td>
</tr>
<tr>
<td>Action: stop and wait until the road is clear</td>
</tr>
</tbody>
</table>

SIMULATIONS

A number of simulation tests have been done. The first simulation consists of a volume of vehicles and a cell configured road-network within a virtual town centre. Figure 4 shows a screen shot of the simulation system that models five hundred vehicles within a virtual town centre illustrating some of the individual vehicles are approaching/entering a roundabout in the town centre. The vehicles must reduce its speed and react to the actions of other vehicles in order to find a safe gap to enter the roundabout. The second simulation consists of a larger number of individual vehicles and the cell-based road-network integrated with a large residential state area.

Figure 4: Vehicles approaching a roundabout at a virtual town centre

Figure 5 shows a traffic flow in a large residential area. Figure 6 shows traffic jam occurrence in the residential area. The system is developed in C++ and DirectX 9 Software Development Kit (SDK) as a graphics language. The simulation is run on a 1000 MHz Pentium processor PC with 256M memory and Matrix G450 graphics card (360MHz, 32M memory). The simulation for the large residential area as has shown in Figures 5 and 6 runs at 10 frames per second, which is near real-time 3D visualisation.

Figure 5: A queue at a traffic light controlled roadway in the residential area.

Figure 6: A traffic jam at a roundabout during morning rush hours of the residential area

CONCLUSIONS

We have developed a prototype of a novel traffic simulation approach using individual-based vehicle models. This approach uses a discrete space for road-network configuration for computational speed, whilst using continuous models for vehicle movement control. A simulation system using such an approach with a 3D virtual environment has been developed, which can provide accurate simulation results of traffic flow situations. The approach described here is effective and the simulation results are realistic. For future work, the system will be fully developed for the analysis of traffic flow, and the evaluations of road designs and the extension to the existing road system, and the rescuing staff training in emergency situations. Further research topics will include traffic management system simulation and evaluation.
REFERENCES


AUTHOR BIOGRAPHY

T. R. Wan is currently a lecturer in the School of Informatics at the University of Bradford, UK. He was appointed as a lecturer at Heriot-Watt University , UK since October 1999 before joining the School of Informatics at the University of Bradford in August 2000. He received his first degree in Engineering and a MSc. with a subject on signal analysis and machine state diagnosis both at the Xian University of Engineering Science and Technology, P. R. China. He has obtained his PhD at the University of Leeds, UK on the subject of image patterns recognition and analysis. Dr. Wan’s research interests are 3D visualization and simulation, Artificial Intelligence modelling and autonomous virtual agent applications, and computer generated imagery techniques. He is the recipient of The Woolmen's Company Silver Medal 1996 (the first prize) by "The Worshipful Company of Woolmen" for the excellence of Innovative Ph.D. research work, London, 1996 He is a guest professor at the Xian University of Engineering Science and Technology. Dr. Wan is a member of IEEE and Eurographics.

W. Tang is currently a senior lecturer in Computer Graphics and Visualisation in the School of Computing at the University of Teesside, UK. She has obtained her Ph.D in Computer Aided Design in 1998 at the University of Leeds, UK. Dr. Tang has been worked as a research assistant at the Department of Electronic Imaging and Media Communications at the Bradford University. She was appointed as a lecture at the Teesside University in 1999 then a senior lecturer. Her research interests are procedure animation techniques, image rendering in animations, and behavioural animations. She has setup and leads the Technical Computer Animation Research Group at the University of Teesside. She is a member of IEEE, and Eurographics.
Analysis of Urban Traffic Using Queueing Networks

Ciro D’Apice, Rosanna Manzo
University of Salerno
Department of Information Engineering and Applied Mathematics
Via Ponte don Melillo, 84084 Fisciano (SA), Italy
email: {dapice, manzo}@diima.unisa.it

Abstract

A simulative traffic model of urban networks based on queueing theory is proposed. The urban network can be represented with an oriented graph: the edges model the roads and the nodes the connection roads. Once the graph modeling the urban section is outlined, some queueing systems are associated with the graph in such way to obtain a queueing network which can be investigated with analytical tools or by means of simulations. An example of application of the simulator based on the described model is presented.

KEYWORDS: urban traffic, modeling, queueing networks, simulation.

1. Introduction

The complexity of the urban traffic and the related problems lead to great efforts in the development of new and adequate models.

Different kinds of techniques and analytical or simulative models have been realized and applied in the area of urban street traffic in order to plan and to improve urban traffic systems. The technique or model choice depends strongly on the needed level of detail and on the problem we aim to solve.

Various simplifications are allowed in analytical models, in order to reduce the system complexity, to have a manageable computation, to reduce or ignore eventual uncertainties which can occur. These disadvantages can be overcome using simulation based models which offer a good tool to analyze performance of the overall traffic network taking into account the system complexity and uncertainties (Chabrol 2000; Schulze 1997).

Existing modeling approaches can be classified in microscopic and macroscopic. Macroscopic approaches describe the traffic flow in the entirety of all vehicles (Knehne 1995). Microscopic models are characterized by the description of single vehicles and relations between them (Lieberman 1991).

The aim of this work is to describe a traffic model of urban networks based on queueing networks (Newell 1982). This model has been used to realize a simulator, by means of a package implemented in Mathematica and able to simulate queueing networks (D’Apice et al. 1999; D’Apice et al. 2000a; D’Apice et al. 2000b).

The queueing networks simulator is able to analyze every system which can be represented as a queueing network.

The article is structured as follows. In the second section a detailed description of the traffic model of queueing networks is presented with particular care to intersections. In the third section, after a brief description on the queueing networks simulator, a study case is reported.

2. Model

A city urban section can be described by means of an oriented graph in which the edges represent the one-way roads and the nodes the intersections.

As we can see in Figure 2.1, to model a two-way road we have to associate an edge with each transit way. In the case of a road with more lanes we have to associate an edge to every lane or to regard the road as a road with one lane but with a greater capability.

Once the graph of the analyzed urban section is outlined, some queueing systems are associated with the graph in such way to obtain a queueing network
which can be investigated with analytical tools or simulations. We have to associate with each node, representing an intersection, a number of queuing systems equal to the number of edges incident with the node. The queuing systems model the time needed to cross the intersection and they are without buffer.

The queuing systems are characterized by the following parameters: the arrivals distribution, the buffer size, the code discipline, and the service times distribution.

The code discipline is of type FIFO (First In First Out).

The process of arrivals is determined by the output of the queuing systems which represent the roads incoming into the node.

We suppose that the service times distribution is exponential. The service mean rate depends on the priority of the roads incident with the node.

For example, let us model the intersection shown in Figure 2.2 where road 1 has priority over road 2. The intersection is modelled by means of two servers, one describe the intersection crossing of vehicles coming from road 1 and the other the intersection crossing of vehicles coming from road 2. Since road 1 has priority over road 2, the transition frequency of the related server is greater, it means that the mean number of vehicles coming from road 1 and crossing the intersection in the unit time is greater than the number of vehicles coming from road 2.

The transit time of roads is modelled by queuing systems associated with the roads. If the server which describes the intersection is busy, the servers modeling the road, once the service is completed, are blocked. All the blocked servers of the road segment represent a waiting queue for the server at the intersection. Since these servers are busy, they don’t allow to other vehicles to go in the road and thus the maximum number of vehicles on it is equal to the number of free servers.

Routing probabilities are used to express the preference of the vehicles in the choice of the directions to bring. A complete representation of the queuing systems set, needed to model an intersection, is shown in Figure 2.3, where the circles represent the servers (one for each edge incoming in the intersection), and the arrows are the possible directions.

Two different road models are described.

The first is represented in Figure 2.4, where the last circle represents the terminal server which belongs to the intersection. As we can see the road is modelled as a queuing system with m servers and without waiting queue. This scheme is similar to the scheme used to model the delay lines with the difference that the number of servers is finite. In fact, in this case, when the road is completely busy, the system is blocked. The servers number coincides with
the number of vehicles which can be simultaneously present on the road. Every vehicle arriving to this road from an intersection occupies a server if there is one available. The service time models the time that the vehicle requires to cross the road. Once the service time is finished, if the intersection is busy, the server is blocked, so it works as a queue for the intersection subtracting a free place in the road.

Figure 2.4: A first model of road.

In order to determinate the service time distribution we consider the road transit time expresses by

\[ t = \frac{l}{v}, \]

where \( v \) is the velocity of the vehicle and \( l \) is the road length.

Regarding the velocity \( v \) as function of the number \( n \) of vehicles in the urban segment, i.e. the number of busy servers, we assume that

\[ v = (v_{\text{max}} - v_{\text{min}}) \left( 1 - \frac{n}{m} \right) + v_{\text{min}}, \]

from which the transit time is given by

\[ t = \frac{t_{\text{min}}}{1 - \frac{(v_{\text{max}} - v_{\text{min}}) n}{v_{\text{max}} m}}. \]

If the velocity is regarded as an aleatory variable, it is possible to express the transit time as

\[ t = \frac{t_{\text{min}}}{1 - \text{Random}[\frac{(v_{\text{max}} - v_{\text{min}}) n}{v_{\text{max}} m}]} . \quad (2.1) \]

In the second road model, as shown in Figure 2.5, the urban segment is divided in various little segments of size equal to the medium length of the vehicles, each of one is seen as a queueing system. These latter consists of a server without buffer and a service time that depends on the real time that vehicles require to cross the urban segment that the queueing system models. The mean time \( t \) spent by a vehicle to cross the various segments of road depends on the aleatory velocity of the vehicle. It is possible to see this velocity as a function of the occupancy state of the servers that precede the segment. If the servers are busy, the velocity is smaller, instead if they are free is bigger.

Finally we have to model vehicular flows which come from the outside or go towards the outside of the section of the urban network. The incoming flows can be described as Poissonian flows, modifying the queueing systems of those roads that communicate with the outside. In the first road model, the incoming flow will be modelled as a queueing system of type \( M/M/m \) where the first \( M \) denotes the effective presence of a Poissonian flow. In the second model, instead, an arrival flow is present only in the first of the servers of a road. In order to model the flows coming from the network, we suppose that the intersection communicate in the outgoing sense. So, denoting with \( p_{ij} \) the routing probabilities related to the system \( i \) of an intersection, the condition \( \sum_{j \neq 0} p_{ij} < 1 \), (where the node 0 denotes the outside), assures that part of the flow goes outside the network.

3. Study case

The model described in section 2 has been used to implement a discrete event simulator. It has been realized in Mathematica and it is able to simulate queueing networks with arbitrary distributions. The input parameters of the simulator are the following:

- the list of the queueing systems modeling the network;
- the routing probabilities matrix that defines the network topology;
- the number of iterations;
- the required outputs (average length of the queue, average system time, etc.).

The simulator allows to:

- add a road defined by the length, the lanes number, the incoming roads, the vehicles velocities, the arrival rates, etc.;
- delete a road;
- insert the probability a vehicle coming out from a road goes in another incident road;
- modify a road (the road name, the length, the lanes number, etc.)
- visualize the created urban network.

In what follows we report some simulation results obtained simulating the traffic in an urban network section of Salerno city, in Italy. As it can be seen in Figure 3.1, the section includes two intersections: the first between Via Cilento and Corso Garibaldi and the second between Via Cilento and Lungomare Trieste. The Roman numbers indicate the points in which the incoming flows to the roads are modelled.

![Diagram of the urban section of Salerno city](image)

Figure 3.1: Urban section of Salerno city.

The length of the road segments and the number of lanes are:

1. Via Cilento Adolfo [2 lanes, 0.22 km]
2. Corso Garibaldi [3 lanes, 0.42 km]
3. [2 lanes, 0.028 km]
4. Lungomare Trieste [2 lanes, 0.59 km]

The overall several-lanes road is modelled as a queueing system with more servers in parallel. The presence of more lanes is modelled assuming that the number of servers of the queueing systems which describe the intersection is equal to the number of lanes of the road to which the queueing system is referred.

The servers’ number of the queueing system which model the road is given by

$$m = \frac{n L}{l},$$

where $L$ is the length of the road segment, $n$ is the number of lanes, and $l$ is the average length of vehicles, which is regarded equal to 7 meters. This value is usually used in traffic simulation and it takes into account the different sizes of vehicles and the security distance.

The incoming flows are regarded as Poissonian flows with parameters:

1. $\lambda = 0.5$;
2. $\lambda = 1$;
3. $\lambda = 1$.

The simulated queueing network is constituted by 8 nodes: 2 nodes for intersection $\alpha$, 2 nodes for intersection $\beta$, and 4 nodes for the roads.

In order to compute road transit time, formula (2.1) is used with the following maximum velocities for the various roads:

1. $v_{\text{max}} = 40 \text{ km/h}$
2. $v_{\text{max}} = 50 \text{ km/h}$
3. $v_{\text{max}} = 30 \text{ km/h}$
4. $v_{\text{max}} = 50 \text{ km/h}$.

We supposed $v_{\text{min}} = 1 \text{ km/h}$.

The crossing times of intersections are exponentially distributed with parameters:

- $\alpha_1$:
  - $\lambda = \frac{1}{6}$
- $\alpha_2$:
  - $\lambda = \frac{1}{3}$
- $\beta_3$:
  - $\lambda = \frac{1}{4}$
- $\beta_4$:
  - $\lambda = 1$.

We report the list of the queueing systems for the various nodes:

1. $M/G/62/0$  $\alpha_1$, $\infty/M/2/0$
2. $M/G/180/0$  $\alpha_2$, $\infty/M/3/0$
3. $\infty/G/8/0$  $\beta_3$, $\infty/M/2/0$
4. $M/G/168/0$  $\beta_4$, $\infty/M/2/0$

where the symbol infinity denotes the absence of an ingress.

Finally we assume the following values for the pairs $\{\text{node}, \text{routing probabilities}\}$:
1. $\{\alpha_1, 1\}, \alpha_1, \{3, 0.1\}, \{-1, 0.9\}$
2. $\{\alpha_2, 1\}, \alpha_2, \{3, 0.1\}, \{-1, 0.9\}$
3. $\{\beta_3, 1\}, \beta_3, \{-1, 1\}$
4. $\{\beta_4, 1\}, \beta_4, \{-1, 1\}$

where $-1$ denotes the outcome from the network.

In Figure 3.2 some simulation outputs are reported: the flow in the node A (Figure 3.1) and the occupancy percentage of roads.

![Figure 3.2: At left: Absolute frequencies of inter-arrival times for vehicles coming to the node A. At right: Occupancy percentage of roads.](image)

4. Conclusions

We described an urban traffic model based on queueing networks that, by means of simulation, allows to analyze urban traffic and in particular conditions of interrupted traffic as those which occur at intersections. The implemented queueing network simulator allows to define city sections and to analyze traffic conditions. It offers the advantage to simulate every system which can be represented as a queueing network.

References


Author Biography

CIRO D’APICE was born in Castellamare di Stabia, Italy and obtained Ph.D. degrees in Mathematics in 1997. He is assistant professor at the Department of Information Engineering and Applied Mathematics in University of Salerno. His research interests include: computer aided learning, queueing theory, hybrid systems, homogenization problems, spatial behaviour for dynamic problems. His e-mail address is dapice@diima.unisa.it.

ROSANNA MANZO was born in Polla, Salerno, Italy. She graduated in Mathematics in 1996. She is Ph.D. student in Information Engineering at the Department of Information Engineering and Applied Mathematics in University of Salerno. Her research areas include queueing theory, self-similar processes, computer aided learning. Her e-mail address is manzo@diima.unisa.it.
SIMULATION ANALYSIS OF AGV SYSTEM PERFORMANCE IN CONTAINER PORT TERMINALS

Martin Grunow
Hans-Otto Günther
Axel Ehren
Matthias Lehmann
Department of Production Management
Technical University Berlin
Wilmsdorfer Str. 148
D - 10585 Berlin
Germany
E-mail: m.grunow@ww.tu-berlin.de

KEYWORDS
AGV dispatching – seaport container terminals – on-line and off-line dispatching mode – simulation modelling

ABSTRACT
A key issue in automated seaport container terminals is the assignment of transportation orders to automated guided vehicles (AGVs). This paper adapts some of the priority rule based approaches known from manufacturing and warehouse applications to cover specific features arising in seaport container terminals. In addition, extended priority rule based dispatching heuristics are proposed, which exploit the capability of the AGVs of carrying one 40 ft or two 20 ft containers at a time. In a simulation study, the performance of these heuristics is compared against a more elaborate pattern based heuristic dispatching algorithm which has been developed by the authors. The latter one can be regarded as an off-line approach, since a complete tour covering a number of pick-up and drop-off operations is generated for each vehicle getting available during a given look-ahead horizon.

INTRODUCTION
This paper is concerned with logistics operations in seaport container terminals. Clearly, this application area contains a number of logistics control issues which are extremely complex in nature (Steenken et al. 2004, for an overview). Hence, sophisticated control strategies are required in order to meet the desired performance measures. As a result, quantitative methodology has received considerable attention to analyze and support the design, operations, and control issues arising in automated container terminals ( Günther and Kim 2004).

Today there is an ongoing trend in the development of seaport container terminal configurations to use automated container handling and transportation technology, particularly, in countries with high labor costs. Hence, manually operated cranes have been replaced by automated ones and often automated guided vehicles (AGVs) are used instead of manually driven carts. Nevertheless, for transportation between different terminals at one location, as is the case in the city of Busan (Korea), conventional trucks are still the primary mode of transportation (Koo et al. 2004). For intra-terminal operation, dual-load AGVs represent a recent development in transportation technology. Such vehicles offer the advantage of being able to transport two 20 ft containers or one 40 ft container at a time. So far, in automated container terminals dual-load AGVs are operated in single-carrier mode, but shall be used as dual-load carriers in the future. Another recent development is represented by so-called automated lifting vehicles (ALVs) which, in contrast to AGVS, are capable of lifting a container from the ground by itself (Vis and Harika 2004; Yang et al. 2004). However, despite their superior handling capabilities ALV systems have not been realized in automated container terminals so far.

DISPATCHING AGVS
The AGV dispatching problem in automated container terminals consists of finding an assignment of vehicles to transportation orders, taking into account locations and target times for pick-up and delivery. In the case of dual-load AGVs, also the size of the containers is of importance. The general goal is to minimize total lateness of AGVs. To solve the AGV dispatching problem at hand, four different approaches can be utilized. While the first one is applicable only for scheduling single-load carriers, the subsequent ones explicitly exploit the increased loading capacity of multi-load vehicles.

The first approach employs basic dispatching rules which are already known from manufacturing and warehouse applications (Egbelu and Tanchoho 1984; Hwang and Kim 1998; Klein and Kim 1996; Lim et al. 2003). These rules are designed for the case of single-load vehicles. Presumably, the most popular dispatching rule is the nearest vehicle rule, which assigns the vehicle which is located closest to the pick-up location of a transportation order whenever a new transportation order is initiated. This rule is often combined with the first-come-first-served (FCFS) strategy which is applied to prioritize waiting transportation orders in case no idle vehicle is available. Another adequate dispatching strategy is represented by the least-utilized-vehicle rule.
combined with the maximum-outgoing-queue (MOQS) size rule, which aims to prevent overflow of output buffers.

The second approach employs extended priority rules to make use of the potential benefits of multi-load AGVs (Bilge and Tanchoco 1997; Lee et al. 1996; Nayyar and Khatore 1993). If, for instance, the vehicle is not fully loaded, then decisions have to be made whether or not the vehicle should stop at one or more intermediate stations along its route (or even deviate from its scheduled route) and make additional pick-ups or deliveries. Hence, pick-up and drop-off rules are required which prioritize the requests from intermediate stations and determine the sequence of pick-up and drop-off operations within the vehicle’s tour. Popular load pick-up rules are first-serve-next-station, which gives priority to the next station on the immediate route, and serve-same-destination, which aims to combine loads from different pick-up locations to be delivered to the same destination. For sequencing the drop-off operations of a fully loaded vehicle, the nearest destination rule and the first-in-first-out rule can be applied. If, on the other hand, no transportation order has been assigned to a vehicle, then dispatching rules similar to the ones for single-load carriers can be used.

As a third approach the pattern-based heuristic dispatching algorithm proposed by Grunow et al. (2004) can be applied. Their approach is based on the distinction of different degrees of AGV availability and the definition of assignment patterns which allow complete alternative AGV tours comprising a sequence of pick-up and drop-off operations to be generated with very limited computational effort. This heuristic dispatching algorithm has been laid out for application in a highly automated seaport container terminal where multi-load AGVs can carry either two 20 ft containers or one 40 ft/45 ft container.

Finally, the AGV dispatching problem for multi-load carriers can be formulated as a mixed-integer linear programming (MILP) model which offers optimal solutions for a finite planning horizon under the assumption of complete information about all future events (Grunow et al. 2004). This approach, however, requires significantly increased computational effort which may be too large in a real container port application.

ON-LINE AND OFF-LINE DISPATCHING MODE

In dispatching AGVs in container ports, the stochastic nature of the handling systems must be accounted for. In real life terminal operations, for example, the sequence of handling operations is regularly altered due to short term decisions of quay crane operators, who may decide to deviate from the pre-determined discharging schedule based on their individual judgment of the specific unloading situation. Moreover, operation times are affected by external factors, like weather conditions or the state and type of the containers. This is particularly true for manual operations, such as quay crane control. Also, transportation times differ due to congestion effects in the AGV system.

Because of these uncertainties, decisions must be made without complete knowledge of the future events. One option to deal with the stochastics of the logistics system is to employ on-line dispatching. According to this dispatching mode, a decision is made when needed and immediately executed (Fiat and Woeginger 1998; Sgall 1998). In this case, no predictive plan is generated. The schedule rather results from a sequence of on-line decisions which are made one at a time as the system status changes (Sabuncuoglu and Bayiz 2000). Dispatching rules for single as well as multi-load carriers, such as the rules described in the previous section, can be used for on-line dispatching. While the application of these rules is simple, their inherent myopic and greedy nature may sacrifice their performance.

In contrast, off-line dispatching requires decisions to be made simultaneously for all transportation orders occurring within a short-term look-ahead period. Thus, a predictive schedule is constructed. However, due to the uncertainty of the future events the schedule may have to be revised when significant deviations occur. This type of planning approach is therefore also termed reactive planning (Sabuncuoglu and Bayiz 2000).

Depending on the factors which trigger rescheduling, the following policies can be distinguished:

- **Periodic**: Rescheduling takes place after predefined time intervals (rolling time horizons; Church and Uzsoy 1992).
- **Event**: Rescheduling is carried out on significant deviations from the current schedule. But also specific events, such as arrival of a new job, may cause rescheduling (Smith 1994; Viera et al. 2003).
- **Hybrid**: A combination of the above policies is applied (Church and Uzsoy 1992).

In our investigation, we use an event-based policy for the off-line pattern-based heuristic. Rescheduling is carried out when a new transportation order is released within the planning horizon or when an AGV becomes available.

**SIMULATION STUDY**

**Design of the simulation model.**

To evaluate the effectiveness of the various dispatching approaches a comprehensive simulation study was conducted. A discrete event based simulation model has been developed using the eM-Plant 4.6.3 simulation system. For modeling a real logistics system through simulation, a major issue in the design of the simulation model refers to the definition of the system boundaries. We decided to build up the simulated system around an AGV guide path and a fleet of vehicles which transport containers between quay cranes located at the berth side and automated stacking cranes which operate at the different storage blocks arranged at the opposite side of the guide path. Thus, sub-systems not included in the simulation model are, for instance, the stowage of vessels, the assignment of containers to locations inside
the storage blocks, the interface to the hinterland, and the traffic control of the AGV system.

In order to simulate automated container terminals of different size, a basic module was defined which constitutes the building block of a flexible terminal configuration (see Figure 1). Hence, by combining various modules a larger terminal configuration can be generated. The basic module consists of four elements: (1) the AGV guide path laid out as a four-lane unidirectional loop, (2) a fleet of AGVs, (3) a single quay crane, and (4) two storage blocks with two automated stacking cranes each. In an optional module design, one or two of the storage blocks are omitted. This design is used to generate a terminal configuration with a series of storage blocks concentrated in the centre of the storage yard. Figure 2 displays a large-sized terminal configuration generated from a number of basic modules. All cranes in the system are linked by a unidirectional mesh-type guide path in which only the traversals between the quay side and the storage yard show a bidirectional orientation.

- Transportation orders are generated according to the working cycle of the quay cranes. For each quay crane the storage block, from where an export container is to be picked up and to where an import container is to be delivered, is randomly selected.

**Experimental scenarios.**
The scenarios investigated reflect realistic terminal environments and consider stochastic variations in the timing and processing of loading and unloading operations of containers. Throughout the numerical experiments, the degree of stochasticity is varied so that the relative performance of the various dispatching methods can be assessed. Each scenario is characterized by the number of quay and stacking cranes in the terminal configuration and the stochastic variations of the handling time per container. All the detailed data required to feed the simulation model (e.g. arrival times of vessels, container and equipment attributes etc.) were generated according to the guidelines of Hartmann (2004) which were derived from the simulation project of a modern automated container terminal. Specifically, we generated a number of scenarios by varying the following experimental factors:

- A small and a large terminal configuration were generated consisting of 4 and 8 quay cranes, as well as 6 and 12 storage blocks (see Figure 2).
- The average number of AGVs per basic module (see Figure 1) was set to 2 and 2.5, respectively.
- Low, medium, and high degree of stochasticity was simulated by considering the handling time per container (loading or unloading) at a quay crane as random values being drawn from the normal distribution with mean $h$ and standard deviation $p-h$, where $h$ denotes the regular handling time of one minute and parameter $p$ was set to 0.3, 0.6, and 0.9, respectively. In addition, the purely deterministic case served as a benchmark.
- Vehicles were operated alternatively as single and dual-load carriers. In the latter case, the capability of the vehicles of loading one 40 ft or two 20 ft containers at a time was utilized.

For each scenario, simulation experiments were repeated five times with different randomly generated input data. The simulation period covered a 24 hour working day with the first 2.4 hours considered as warm-up period. On average 215 transportation orders were generated per day and per quay crane. The travel time of an AGV was set at 25 seconds between two adjacent quay cranes and 20 seconds for the traversal between the quay side and the storage yard side (see the layout indicated in Figures 1 and 2). The availability of an AGV is determined by its status after finishing the current trip. A single-load AGV is considered available during its trip to the drop-off location. Dual-load vehicles are fully available during the trip to their last drop-off location and partially available during the trip to the first pick-up location of a 20 ft container or to the first drop-off location. Of course, both types of vehicles are considered available when parked idle at some dwell points in the guide path.

Moreover, our simulation model is based on the following major assumptions:

- The loading and unloading sequence of containers is known for each vessel.
- Travel times of vehicles are assumed to be deterministic.
- The time required by a quay crane for loading and unloading a container is considered as stochastic.
Dispatching requests are generated, when a new transportation order is released (transportation order initiated dispatching) or an AGV becomes available (AGV initiated dispatching). Two on-line and one off-line dispatching rules were implemented.

- **On-line dispatching**: (1) NV/FCFS: nearest vehicle (for transportation order initiated dispatching) / first-come-first-served (for AGV initiated dispatching), (2) LUV/MOQS: least utilized vehicle (transportation order initiated dispatching) / maximum-outgoing-queue size (AGV initiated dispatching)
- **Off-line dispatching**: Pattern-based approach (cf. Grunow et al., 2004).

Each of these approaches is used as single-load and dual-load dispatching method, respectively. If a transportation order is assigned to a partially available dual-load carrier, the on-line rules schedule its pick-up operation after the pick-up of the order already assigned to the vehicle. The sequence of the drop-off operations is determined according to the scheduled drop-off times. The on-line heuristics are myopic. For transportation order initiated dispatching, they include only the transportation order that follows the order that has just been picked up by an AGV at a quay or stacking crane and will next be made available by the same quay or stacking crane. Accordingly, the on-line heuristics make only a single 1:n or an m:1 assignment at a time. The previous schedule is maintained and merely expanded.

In contrast, the pattern based off-line approach covers all transportation orders within a short-term look ahead period (7 minutes in our experiments). It allows reassignment of all operations apart from the one to which the vehicle is currently en route (to avoid deviations) and apart from the drop-off of already picked-up containers (which clearly must be done by the AGV currently transporting the container). Independent of the triggering event of the dispatching request, a new m:n assignment between unassigned orders and currently available vehicles is generated iteratively. For the unassigned order with the earliest release time the assignment with the least total lateness is determined. For this purpose, all available vehicles and all feasible assignment patterns (i.e. sequences of the operations of a vehicle) are evaluated. The procedure terminates, when all orders are assigned or no available vehicle is left.

**RESULTS AND CONCLUSION**

As a general result, we found that the performance of the small-sized terminal configuration is in accordance with the one of the large configuration. The numerical results may be summarized as follows:

- As expected, the performance of all approaches diminishes with an increasing degree of stochasticity.
- The LUV/MOQS rule is not an expedient strategy for dispatching AGVs in automated container terminals. Contrary to manufacturing and warehouse applications from where this rule originates, no buffers exist. Thus, the arrival times of AGVs at the quay and stacking cranes are of utmost importance to avoid idleness of the handling equipment, yet disregarded by the LUV/MOQS rule.
- Even for the scenarios with high stochasticity, the pattern based heuristic dispatching algorithm performs better than the NV/FCFS rule (with respect to the total lateness of transportation orders).
- There is no significant deviation in the performance of dual-load and single-load AGVs when the number of AGVs per basic module is large. In this case, only a single container is assigned to a vehicle despite its larger capacity, i.e. it is actually used as a single load vehicle.
- The basic on-line rules do not dispatch the multi load carriers appropriately and, thus, should not be used when the AGV capacity is scarce.

Our simulation experiments show that the pattern based off-line dispatching approach is superior to traditional rules used in manufacturing environments. Currently, we are conducting extended simulation experiments. Moreover, we attempt to further improve the pattern based approach by identifying substantial schedule deviations on which rescheduling is triggered.

**REFERENCES**


ACKNOWLEDGEMENT

The authors greatly appreciate the support from the German Academic Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under grant No. GU 459/4.

AUTHOR BIOGRAPHIES

MARTIN GRUNOW is Assistant Professor at the Technical University Berlin. He received his PhD from the same institution before working in the research department of degussa, a multinational producer fine chemicals. He has published in several refereed journals such as the International Journal of Production Research, International Journal of Production Economics, and OR Spectrum. For the latter he also acts as an Assistant Editor. His research interests include the application of simulation and optimization methodology to container ports logistics and the coordination of production networks in the process industries. Another focus of his research is the production of electronics assemblies.

HANS-OtTO GÜNTHER is Professor of Production Management at the Technical University Berlin. He received his PhD degree from the Free University of Berlin (Germany) and completed postdoctoral studies (Habilitation) at the University of Mannheim (Germany). His papers appeared in journals such as International Journal of Production Research, EJOR, OR Spectrum, Computers in Industry, Production Planning and Control, International Journal of Production Economics. Currently, he is Managing Editor of the journal “OR Spectrum: Quantitative Approaches in Management”. His research interests include Supply Chain Management and Advanced Planning Systems, operations planning and scheduling in the chemical industry, logistics control systems, and printed circuit board assembly.

AXEL EHREN is currently completing his master thesis at the department of Production Management, Technical University Berlin. His research interests include the application of simulation to container port logistics and the optimization of industrial supply networks.

MATTHIAS LEHMANN is PhD student and research assistant at the department of Production Management, Technical University Berlin. He graduated in economic mathematics at the same university. He has published several papers in conference proceedings and refereed journals such as the OR Spectrum. His research interests include automated guided vehicle systems, container port logistics, and online scheduling.
SUPPLY CHAIN SIMULATION
AN INDUSTRIAL SUPPLY CHAIN SIMULATION MODEL

PJ Byrne and Cathal Heavey,
Enterprise Engineering Research Group,
Department of Manufacturing & Operations Engineering,
University of Limerick, Ireland
pj.byrne@ul.ie

KEYWORDS
Supply Chains, Object Orientated, discrete event simulation, decision support system.

ABSTRACT

The following paper presents the formulation and development of a case study supply chain simulation model as developed for an industrial company. The case study company in question is considered to be a vertically integrated organisation, offering a complete range of its related industries products to a global marketplace.

The paper reviews the scale of system being analysed, the type of data required to populate such a model, such as product routings, standard times, work centre capacities and shift cycles and the level of detail included in the study and the performance outputs from the model. These outputs include the percentage of demand that is both On Time and In Full (OTIF%), the days of inventory held in finished stock and also the finished stock quantities.

The paper also presents an experiment carried out on the model to determine the effective trade-off between these performance outputs, from operating three different finish stocking policies. The stocking policies consist of one where all products are make-for-stock (MFS), one where all products are make-to-order (MTO) and one where select products are MFS, based on a number of conditions and the remainder are MTO.

INTRODUCTION

In today’s global markets, fierce competition caused by the introduction of products with short life cycles, and the heightened expectations of customers have forced business enterprises to invest in, and focus attention on, their supply chains, (Simchi-Levi, Kaminsky et al. (2000). Competitiveness of products and services is no longer the responsibility of a single organisational unit, but the supply chain as a whole. Successful supply chains are ones that either have a ‘productivity advantage’ or a ‘value advantage’ or a combination of the two. The management of these supply chains is consequently fundamental to an organisations competitive advantage in industry today.

Supply chain inventory modelling can be broadly categorised into two distinct groups, analytical and simulation inventory modelling. The body of work in relation to analytical modelling is quite extensive with the first work being carried out in this area by Clark and Scarf (1960), where a simple two echelon system was analysed. Numerous extensions to this work can be found throughout the literature, however it can be noted that these models are very restrictive in both size and application, often built around single products, single time periods and single product lines. On the other hand it can be seen in the literature that impressive results have been achieved where supply chain management techniques have been used, such as those stated by Hewlett Packard, (Lee and Billington (1995)), Campbell Soup (Gachon and Fisher (1998)), IBM (Lin, Ettl et al. (2000)) and BASF (Grupp (1998)). However, where simulation has been used for analysing supply chains, it has been on very large systems with simple “black-box” lead-times used. This paper proposes the use of such simulation technology in Small-To-Medium enterprises with additional functionality in the manufacturing function of the supply chain, which includes re-entrant flow, large quantities of finished products (approximately 10,000) with a large number of individual work centres and machines (42 work centres and 276 machines), with high levels of interactions between product families and work centres.

SUPPLY CHAIN CASE STUDY

The organisation involved in the study is classed as a vertically integrated raw material supplier, offering a complete range of its related industries products to a global marketplace. Scharny and Skjott-Larsen (1995) define vertical integration as “the ownership by one organization of other firms in its supply or distribution network... The totally integrated firm is completely self-sufficient. The non-integrated firm is completely dependant on market forces and other organizations for its operations.” Christopher (1998) states, “Vertical integration normally implies ownership of upstream suppliers and downstream customers.”

The companies supply chain structure as shown in Figure 1, can generally be regarded as an extension of the well studied serial multi-echelon supply chain, being supplied by a number of raw material suppliers and supplying a number of distributors, who in turn supply individual customers.

The following section will give a basic description of the overall structure of the supply chain and the type of processes carried out at each point in the system. As already stated the company is vertically integrated and owns and controls almost the entire supply chain across a global spectrum. The exceptions to this are a number of the multiple initial raw material suppliers and a number of the distributors. The raw materials are the basic building blocks necessary to produce the end products, which include items that will be used in the actual product and items that will be used in the manufacturing of those products. These materials are all stored awaiting component manufacture. The component manufacture is
the first step necessary in the production of a finished product. This processing section in the system is controlled using a forecasted production plan and has a number of assembly and machining processes associated with it. Initially the raw materials are amalgamated in the correct quantities related to each individual products desired characteristics, after which the assembled component is machined to its required final quality and dimensions, and finally they are inspected before they are transported to the component storage location.

In the capsule development process components are taken and combined together to create capsules. This is the last stage where items are added together to produce a part. These assembled capsules will form the base parts, which will move through the supply chain being further processed. In this part of the supply chain the components are taken and initially placed in an oven to remove the impurities in the product followed by furnace treatment to remove the remaining impurities. At this point the relevant components to produce the required product are combined into a capsule by means of a welding process. Again this area operates under the control of a forecasted production plan and after the capsules are created they are transported on to the assembled capsules stocking location. There are approximately 60 different products at this stage in the system, in the form of assembled capsules.

After completing the pre-assembly the parts are subjected to a highly controlled pressure treatment. The assembled capsules are placed in a pressure treated environment for a specified length of time after which they are checked and machined to product specific dimensions and quality. After the products have been machined their final quality is checked and they are categorised in relation to both their quality and their dimensions. This area also operates using a forecasted production plan. This process converts each of the assembled capsules into a number of different products as outlined above in relation to quality, dimensions and initial capsule characteristics. There are approximately 150 different products produced at this point, which are then transported to the bulk stock location.

The next section in the supply chain is the bulk processing section of the system. This section of the supply chain contains a number of work centres whose purpose it is to prepare the product for final processing. The parts produced in this section of the system are produced in relation to work orders, which are triggered by reorder points in the buffer stock area. After parts have been processed in this section they are assessed and re-characterised based on their original characterisation, what additional bulk processing has been carried out on them and their assessed quality level. Once the parts have been assessed they are transported into buffer stock. There are approximately 1000 active parts in buffer stock at any point in time.

After buffer stock, parts can move in two directions (see Figure 1) depending on the requirements of the work order. These two areas are the finish processing area and the intermediate processing area. The parts produced in this section of the system are produced in relation to work orders similar to those mentioned in the previous paragraph, which are triggered by reorder points in the finish stock area. There is increased complexity in this section of the system, in that buffer stock itself can also trigger production of parts, which require intermediate processing, where a part is taken from buffer stock, processed and sent back into buffer stock with a new part number, where it must then await a work order to take it through finish processing. This procedure can be seen in Figure 1, with the directional arrows indicating the flow of material. If a work order has been created from finished stock then the processed part, flowing through either processing area is transported to the finished stock location.

The distributors in the system are the driving point for the production in the supply chain, as this is the last point in the supply chain where the company has good visibility. The distributors hold a local stock of inventory, which they use to supply their customers, and it is these distributors that create demand for finished items from the finish stock, thus reducing the finish stock and triggering reorder points to start processing. The distributors send their demand to the company, as they require products according to their own stocking policies. The customers order items from the distributors in smaller quantities in relation to their required amounts.

**SUPPLY CHAIN PROBLEM FORMULATION**

In the previous section the problem of interest was formulated, through meetings and conversations with company personnel, who identified areas to be analysed and the level of detail required from the study, such as the effect of the manufacturing function on the overall supply chain performance. It was also at this stage that key performance measures that were to be used in the model
were identified, which included the OTIF metric, the product velocity metric and the days of inventory. It was also at this point that it was decided that the model developed would be built using an object orientated simulation package, eM-Plant. This package was chosen for a number of reasons. The first being the fact that it is object orientated, which allows for easy replication of objects (such as work centres, etc.). The second reason is its ability to use an ODBC link (Open Database Connectivity Link) to connect the simulation model directly to an information database, and the third main reason was the flexibility offered by eM-Plant in relation to its programming language (SimTalk) to customise objects to realistically resemble the existing physical supply chain. This was especially useful when modelling the manufacturing function and for examining the effect of these functions on the overall supply chain performance.

**SUPPLY CHAIN MODEL INTRODUCTION**

It can be noted from the general overview of the company, the issues the company would like to have analysed and the level of detail incorporated in the supply chain that it would not be possible to model this system accurately and efficiently using analytical techniques. Therefore the use of simulation was decided upon as a viable process for analysing the supply chain. However with simulation it is not possible to model a system exactly, but only to gather data and attempt to estimate the true characteristics of the system. With this approach it is necessary to introduce some simplifying assumptions to the model, which ease the development of the model but do not take from the overall structure of the model or the validity of the results.

A number of key simplifying assumptions, which were regarded as necessary and justified prior to the development of the simulation model were outlined as follows:

- The use of a single processor in each work centre, with a suitably adjusted processing time as opposed to splitting batches across a number of machines, to eliminate the need for batch splitting in work centres, which in turn reduces the number of entities in the model at any one time, thus speeding up the processing time of the model.
- Substitution is used across all products in the company when necessary, but there is no formal procedure for it. Therefore a number of substitution tables were developed for products with high demand to incorporate the concept into the model.
- Daily demand is sent into the model in relation to the earliest required date from the customer.
- Daily demand for same products is sent into the model as a consolidated work order, i.e. if two distributors order the same product on the same date the two orders are consolidated and the work order is given the earliest distributor required date as the due date for the entire work order.
- Demand is only satisfied when the entire order can be fulfilled.

- It is assumed that there will be an infinite stock of material in bulk stock, which is the most upstream stage in the model currently being analysed.

**Data Transfer – ODBC Link**

The simulation model itself is connected to a number of database tables (Figure 2), which are stored in MS Access, by means of an ODBC link between them and eM-Plant. This link enables easy transfer of data from the real system into the simulation model and can be set up to continually update and maintain the information in the model. The databases, which are used in the model, contain information on the individual distributors demand, the product routes for parts as they flow through the system with associated standard times, the processes that produce different products and associated yield factors, the output from these processes and the overall bills of material.

**SIMULATION MODEL OBJECTS**

The following section presents details of the supply chain model, in relation to the overall structure (see Figure 3), development and operation of the model. The steps used for model development are those as described for building a sound simulation model in Law and Kelton (2000). It can be noted that similar key steps and concepts can be found from a number of other alternative sources.

**Distributor’s**

The distributor is an object, which holds information on each distributor in the simulation model in relation to that distributors demand sorted by the order date. This information is imported into the model from a general unsorted MS Access table, which contains the demand information for all distributors using the ODBC link and SQL code written in eM-Plant using SimTalk. The demand information imported is the part numbers, product specific information, the year and period the product was ordered, the date of the order, the customers required date, the date the part was actually dispatched from the system (using historical data) and the quantity of the product ordered.
Demand

On double clicking this, the demand object the user is presented with a dialog box (Figure 4), which presents the user with two tabs (Generation and Demand Table). The Generation tab facilitates the importation (or re-importation) of the distributor demand as described in the previous section. It also enables the generation of the demand profile to be used by the model in relation to the products ordered, when they were ordered, the quantities of the orders and the distributors who ordered the products. The Demand Table tab allows the user to open and view this demand table, as the simulation model will use it (Figure 5). Demand is categorised into daily demand, and each of these daily demand categories contain a list of products ordered on that day sorted by the earliest distributor due date for that product on that day. There is also a table recording the distributors that ordered that product on that particular day and the date the product was actually dispatched from the historical data. The overall customer require date as used in the simulation model is the earliest required date for the product by all distributors ordering that product on that day.

Figure 4: Demand Dialog

The demand object also controls the release of demand into the simulation model, when the simulation model has been started, in relation to the current time of the simulation. At the start of each day the model checks to see if any products have been ordered. If one or more products have been ordered then the model goes about satisfying these orders. It loops through the demand table for that day checking each product in turn. If demand for the product can be met from stock then the demand is satisfied immediately, if not then whatever stock was present is assigned to that order and the remainder backordered and an order is generated.

Figure 5: Period Demand

Stock Locations

The stock location objects store information in relation to the continually changing stock positions in the simulation model. The model contains two fully functional stock locations, which are the finished stock and the buffer stock locations. On double clicking any one of these objects, the user is presented with a dialog box (Figure 6), which presents the user with four tabs (*XEX, Reorder, Stock and Table). The first tab is DXEX for the finished stock location and HXEX for the buffer stock location. On this tab the user has the option to import and view the HC processes for producing EC parts in the case of the buffer stock location and import and view the EX parts necessary to produce DX parts in the case of the finished stock locations.
Figure 6: Stock Locations Dialog

The second function of this tab is to import the relevant routings for both the HC processes and the DX items. The routings contain information on the work centres the part needs to visit and the associated standard time to process that part in each work centre. Both of these functions import this information from the MS Access database through an ODBC link between the simulation model and the database.

The reorder tab contains information in relation to the reorder parameters as used by the simulation model, which includes the reorder. The reorder parameters are determined in relation to the family to which each individual product is categorised. The user has the option of changing the safety stock weeks, the typical batch sizes, and the typical lead times used by the model for calculating the reorder points for the different products. The “generate parameters” button generates the reorder points for the current data in the model. The stock tab enables the model to generate an initial stock level for stock in the stocking locations, and the “generate initial stock” level button facilitates this process. The table tab allows the user to view the information stored in the stock table, an example of which is shown in Table 1. In addition to the reorder point information the table also keeps a record of the quantity of parts in stock, on order and the backorders throughout the entire simulation run. The last two columns in the table facilitate the calculation of the number of days of inventory in stock for that part.

The buffer stock location has an extension to the finished stock location, in that it has an additional waiting table for parts that can't be serviced immediately from buffer stock. So when a demand arrives for a part in finished stock and there is an insufficient quantity of finished product in stock to satisfy this order a work order to produce this part is sent to processing. This work order on arrival into a processing area is then checked to see if there is a sufficient quantity of the parent item (stored in buffer stock) to produce the quantity of these parts. If there is not then the work order is placed into this waiting work orders table along with all the relevant information about the part and its necessary path back down the supply chain and a work order is sent back to the next step in that products supply chain to produce the necessary buffer stock product and the process repeats itself. If this work order also requires a buffer stock item and again it is also not available then a new entry has to be placed into the waiting work orders table and have the point recorded that it is linked to the previous work order. The model has been set up in such a way that when a part arrives into the buffer stock location from one of the processing areas the first thing that is checked is whether or not the order was for a waiting order in the waiting orders table. If it was then that work is then sent to its appropriate processing point with its recently acquired parent material. However if parts enter the buffer stock location areas as bi-products, but they can be used to service some waiting order then the waiting work order is serviced with that material immediately and the waiting work order is removed from the list.

### Table 1: Stock Table

<table>
<thead>
<tr>
<th>Item Code</th>
<th>REQB Item Code</th>
<th>ECID Dec/ct</th>
<th>Qty per</th>
<th>Yield</th>
<th>Family</th>
<th>Order Policy</th>
<th>Safety Stock</th>
<th>Reorder Point</th>
<th>Reorder</th>
<th>Po No</th>
<th>In Stock</th>
<th>Backorders</th>
<th>On Order</th>
<th>Issued</th>
<th>RTO Days</th>
<th>Days of Inv</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCDXXXXX</td>
<td>185</td>
<td>UPR0716-35055</td>
<td>0.06980</td>
<td>0.9</td>
<td>A</td>
<td>MTO</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>752.1739139447</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>DCDXXXXX</td>
<td>215</td>
<td>UPR0716-35055</td>
<td>0.00860</td>
<td>0.9</td>
<td>A</td>
<td>MTO</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>86</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>78.260696950517</td>
<td>96</td>
<td></td>
</tr>
<tr>
<td>DCDXXXXX</td>
<td>215</td>
<td>UPR0716-35055</td>
<td>0.01760</td>
<td>0.9</td>
<td>A</td>
<td>MTO</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>30</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>46.96532173010</td>
<td>72</td>
<td></td>
</tr>
<tr>
<td>DCDXXXXX</td>
<td>2003</td>
<td>RBF1916-12FL</td>
<td>1.00000</td>
<td>0.95</td>
<td>B</td>
<td>MTO</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>72.391304347825</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

### Processing Areas

The processing area objects contain the controls for material movement throughout the simulation model. They hold the work centre objects, which will be explained in more detail in the next section. They also control the movement of material and the generation of work orders in these areas. The processing areas also store and feed the model with the number of machines in each work centre, which can be changed to determine the effect of an addition or reduction of machines in a work centre on the overall performance of the system. Also contained within these areas is a velocity module, which is used to calculate the velocity of work orders in each of the different areas.

There are three processing areas within the model, including bulk processing, intermediate processing and finish processing (See Figure 7 for an example of the bulk processing area). When a demand arrives into a processing area the controls checks on the availability of parent material to manufacture the part. If there is insufficient material a substitute is then searched for.

The processing areas also contain information in relation to process percentages, etc. and when the work order exits its last work centre in the processing area its output is created taking into account this information. The work order is then sent in the correct quantities to the required locations.
Work Centres

The work centres contain the controlling mechanism for the processing of work orders as they move through the supply chain model. They consist of a buffer object, which holds the work orders waiting processing and a single processing unit to process the work orders. In order to simplify the model (to speed up its operation) a single processor was used instead of the actual number of machines (servers) in the work centre. Work orders are stored in the buffer and are processed on a first in, first out basis. When the work order enters the processor the processing time is calculated as the ((Work Order Standard Time * Work Order Qty) / (Servers)), which is equivalent to splitting the batch equally over each machine, which is being used as an assumption in the model. When the part has been processed it is then sent to its next operation on its production route. If this operation is the last operation on its route, it is either sent to the appropriate stocking location, where it is either placed in stock, transferred to the customer or a combination of the two. Also stored in each work centre is the reject quantity produced in relation to the yield factors, the Work In Process items in the work centre and how many orders these WIP items consist off. Each processor in each work centre also has an associated shift cycle as described in more detail in the next section.

Shift Cycles

The shift cycles control the operation time of each work centre in the model and currently there are six different shifts controlling the different work centres. There are three straightforward shifts, which operate on a 1, 2 or 3 shift cycle 5 days a week. There are two shifts set to operate on the numbers of hours per week (there are highly automated machines in these work centres) and these operate for 100 and 130 hrs per week including weekends. Finally there are a number of work centres, which operate on a cyclic control of one shift, and two shifts every other week, which for simplicity was modelled as an average one and a half cycle shift.

Model Outputs

In the model the stats module accumulates and calculates weekly the output parameters for each simulation run and the excel module transfers the models output from the format present in eM-Plant to Microsoft Excel for further analysis. The stats module calculates the stock levels throughout the model, the work centre utilisations and the KPIs, which include the Days of Inventory, the velocity levels and the OTIF product percentages. This information is then exported to Excel, for run time reviews.

EXPERIMENTAL RESULTS

Such a supply chain simulation model provides the capacity for extensive experimentation, to aid decision makers with both structural and policy decisions throughout the organisation. Experimentation has been carried out on such areas as, changing market demand trends, stochastic demand changes, stockpiling policies and the production controls in the work centres. The following section reviews part of one such experiment as carried out to review the effect of changing the companies operations from a stocking policy where all finished stock products were Made-For-Stock (MFS), to a policy where a select group of these products were changed to Make-To-Order (MTO) based on the products demand patterns.

Experimental Description

The experiment was carried out keeping all variables in the model constant with the exception of the stocking policies for each finished stock product. A constant demand set was used for each run of the experiment. The experiment itself was run under three different conditions:
All products are MFS (All products are stored in Finished Stock), which was current practice.

Limit MTO (selected products are stored in Finish Stock as determined by stocking policy rules, set out by the company, based on a four constraints, which include the following product analysis, (1) Pareto, (2) Volatility, (3) Peaky Demand and (4) a Declining Demand), which is the policy being analysed.

All products are MTO (only excess produce is stored in Finish Stock), which can be used as a benchmark for the other two policies.

Using the demand set supplied there are approximately 3,000 stocked products under MFS, 220 under a Limited MTO and 0 on a MTO stocking policy. The purpose of the experiment is to determine the effect of each scenario on the percentage of products that are both On Time and In Full (OTIF), the average days of inventory held in finished stock and the quantity of finished stock held. Although there are many products in the system they can be categorised into 8 different product families, but for this analysis only the top 3 families are analysed. These families have been categorised as families A, B and C. Each of these families have been further divided into Rnds and Segs, because although they use the same parent material they have different processing routes through the supply chain.

RESULTS

On Time In Full (OTIF%)

It can be seen from Table 2 and Figure 10, that the change in stocking policy from All MFS, through a limited MTO to All MTO, caused a reduction in the OTIF%. It can be seen that the overall OTIF% drops from an average of 93.02% to 88.08%, if the stocking policy is to be changed from All MFS to a limited number of products manufactured as MTO, with the B Seg being affected most dropping from 70.4% to 43.8%.

Table 2: OTIF%

<table>
<thead>
<tr>
<th></th>
<th>All MFS</th>
<th>Limit MTO</th>
<th>All MTO</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Rnd</td>
<td>98.0</td>
<td>93.8</td>
<td>88.1</td>
</tr>
<tr>
<td>A Seg</td>
<td>93.1</td>
<td>90.6</td>
<td>90.9</td>
</tr>
<tr>
<td>B Rnd</td>
<td>88.9</td>
<td>79.1</td>
<td>78.0</td>
</tr>
<tr>
<td>B Seg</td>
<td>70.4</td>
<td>48.8</td>
<td>37.6</td>
</tr>
<tr>
<td>C Rnd</td>
<td>99.3</td>
<td>98.1</td>
<td>98.1</td>
</tr>
<tr>
<td>C Seg</td>
<td>84.9</td>
<td>85.3</td>
<td>81.4</td>
</tr>
<tr>
<td>Totals</td>
<td>93.02</td>
<td>88.08</td>
<td>84.93</td>
</tr>
</tbody>
</table>

Average Days of Inventory

It can be seen from Table 3 and Figure 11, that there is significant reduction, in the range of 15 to 25 days in the average days of inventory in finished stock across each product family as the stocking policy moves from all items MFS to limited MTO.

Table 3: Average Days of Inventory

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week</td>
<td>All MFS</td>
<td>Limit MTO</td>
<td>All MTO</td>
</tr>
<tr>
<td>13</td>
<td>35</td>
<td>12</td>
<td>41</td>
</tr>
<tr>
<td>26</td>
<td>35</td>
<td>11</td>
<td>42</td>
</tr>
<tr>
<td>39</td>
<td>37</td>
<td>11</td>
<td>43</td>
</tr>
<tr>
<td>52</td>
<td>37</td>
<td>11</td>
<td>44</td>
</tr>
</tbody>
</table>

Finished Stock

It can be seen from Table 4 and Figure 12, that there is a significant drop in the quantity of finished stock held as the model stocking policy changes from all items MFS to limited MTO, with an average reduction of 24,000 products.
**Table 4: Finished Stock**

<table>
<thead>
<tr>
<th>WEEK</th>
<th>All MFS</th>
<th>Limit MTO</th>
<th>All MTO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>35472</td>
<td>14879</td>
<td>5661</td>
</tr>
<tr>
<td>13</td>
<td>31869</td>
<td>13191</td>
<td>4819</td>
</tr>
<tr>
<td>26</td>
<td>32405</td>
<td>13099</td>
<td>4570</td>
</tr>
<tr>
<td>39</td>
<td>31028</td>
<td>12378</td>
<td>4296</td>
</tr>
<tr>
<td>52</td>
<td>31769</td>
<td>12788</td>
<td>4264</td>
</tr>
</tbody>
</table>

**Figure 12: Finished Stock – MFS – MTO**

**DISCUSSION**

One of the major obstacles encountered while carrying out this study was the poor quality of information obtained from the company due to poor information storage standards, which required significant resources to correct. However, taking this into account, the detailed supply chain simulation models can be developed in a relatively short period of time (estimated between 3 and 6 months), and can be executed in a relatively short period of time given the level of detailed associated with the simulation, in the case of the current model 3 minutes and 17 seconds to complete one run of the model. Assuming the model has already been populated with such things as products, standard times and bills of materials.

**CONCLUSIONS**

It was clear to see from the experiment that there was a significant difference in the results obtained under these three policies in relation to the OTIF percentages, the average days of inventory in finished stock for each product family and the average finish stock levels. As an example of this view Table 5 for product family A. It can be seen from this table that the OTIF percentages was 97.97% for A Rnds, with an average of 38 days inventory being held in finished stock for product family A, while having an average of 32,509 items stored in finished stock across all products while using a MFS stocking policy. When the stocking policy was changed to a limit on MTO products the OTIF percentages fell to 93.83% for A Rnds with an average of 14 days inventory being held in finished stock for product family A, while having an average of 13,267 items stored in finished stock across all products. Taken this information into account the company can make an informed decision in relation to the stocking policy to be used. It is clear to see from this analysis that there is a trade-off between the On Time In Full achievable and the quantity of stock held in finished stock, and this experiment clearly indicates the level of this trade-off. Further to this experiment it is envisaged that further experimentation will be carried out to review the MTO policy itself. Such models can greatly improve Small-To-Medium supply chains, helping them to understand the effects of parameter changes on overall supply chain performance indices. Further experiments have been carried out (not reviewed here) to investigate the effect of the manufacturing function (safety stock rates, additional work centre capacities and buffer restrictions) on the supply chain performance parameters and to review the effect of demand changes caused by market trends.

**Table 5: Results Summary**

<table>
<thead>
<tr>
<th>OTIF% for A Rnds</th>
<th>Avg Days in Finished Stock for A</th>
<th>Total FReEs</th>
</tr>
</thead>
<tbody>
<tr>
<td>All MTO</td>
<td>85.14</td>
<td>11</td>
</tr>
<tr>
<td>limit MTO</td>
<td>93.83</td>
<td>14</td>
</tr>
<tr>
<td>All MFS</td>
<td>97.97</td>
<td>38</td>
</tr>
</tbody>
</table>

**REFERENCES**


**BIOGRAPHY**

**PJ BYRNE** is currently working as a research fellow, as part of the enterprise engineering research group in the University of Limerick, Ireland since Jan 2004. Prior to this he completed a PhD in Engineering in the Manufacturing and Operations Engineering Department in the University of Limerick.

**CATHAL HEAVEY** is a Senior Lecturer in Operations Management in the Department of Manufacturing and Operations Engineering in the University of Limerick, Ireland. He also currently holds the position of assistant dean of research for the college of engineering in the University of Limerick.

Sana KOUBAA, Emma GARGOURI and Slim.HAMMADI
Ecole Centrale de Lille, LAGIS
UPRESA 8021, BP48, Cité Scientifique 59651
Villeneuve d’Ascq CEDEX– France
E-mail : sana.koubaa@ec-lille.fr

KEYWORDS
Agro-food Industries, Evolutionist Algorithm, Scheduling, Supply Chain Management.

ABSTRACT
Improving supply chain management is very important to face the globalisation challenges especially for agro-food industries. This paper presents an evolutionist algorithm based approach to resolve one of the objectives of SCM which is to synchronize production and distribution function in the aim to manage at time customers orders while considering the constraints of the production and those of the distribution, in the agro-food industry.

INTRODUCTION
The globalisation combined with the markets instability supposes that industries have a strong capacity of ability to react without degrading its global performance level. This challenge cannot be met effectively only by change within specific organizational units, but rather critically depend on the relationships and interdependences between different organizations, both internal and external to a given industries.

However, many industries, in particular those of the agro-food sector, always have a vision split up with the production and information flows. This view looks for the local optimum of each entity in the industry without optimising the global performance.

To resolve this problem, the agro-food industries have to make effort to manage and control the manufacturing organisation taking into account the specificities and the particularities of the agro-food supply chain (Gargouri et al. 2001) (Gargouri et al. 2002). Indeed, the welfare of any entity in the supply chain directly depends on the performance of the others, along with their willingness and ability to coordinate. This is what supply chain management (SCM) is all about.

The task of SCM is to deploy resources across a supply chain to produce high-quality goods as inexpensively as possible and when the customer wants them. It governs decisions such as which suppliers should be used for which products, in what order products should be manufactured, when new jobs should be started, and when new orders should be placed. Efforts to describe explain and resolve SCM problems have recently led to a plethora of research and writing in this field (Ganeshan et al. 1998). At the same time, the level of attention SCM now receives in business practices also heavily influences the growing interest in SCM research.

In this paper, we propose an approach to resolve one of the objectives of SCM which is to synchronize production and distribution function in the aim to manage at time customers orders while considering the constraints of the production and those of the distribution. Indeed, we are interested in the resolution of the scheduling problem of the production tasks on the conditioning post by taking into account constraints of distribution, in the agro-food industry.

The rest of this paper is organized as follows: Section 2 briefly describes the scheduling problem we are addressing and the requirements we identify; Section 3 describes the evolutionist algorithm for supply chain management in agro-food industry; Section 4 present some simulation results; Section 5 gives concluding remarks and perspectives.
THE SCHEDULING PROBLEM

On the distribution side, we consider $m$ turned $T_k$ $(1 < k < m)$ having each a wished delivery date $d_{liv_k}$, a duration averages $DT_k$ and a possible delay $R_k$ engendered with the corresponding lorry.

Every tour is formed by $n_k$ command $c_{jk}$ $(1 < j < n_k)$ supposed firm and compound each of a definite number of finished products to be made.

We define, for every finished product $i$ of a command $c_{jk}$, the quantity $q_{ijk}$ to make, the date of the earliest manufacture beginning $r_{ijk}$ and the date of the latest manufacture end $d_{ijk}$.

On the production side, we associate the conditioning post with a set of operations to be authorized:

$$O_{rcdo} = \{Op_i(Q_i), i \in I\}$$

Where $Q_i$ is the sum the quantities of all the finished product commands, having as reference index $i$, in all the tours $T_k$ defined on the scheduling horizon:

$$Q_i = \sum_{k=1}^{m} \sum_{j=1}^{n_k} q_{ijk}.$$ 

The considered scheduling problem consist to give the execution order of the elementary operations $Op_i$, on the conditioning post, to satisfy cleaning operations constraints, delivery delays constraints and the particular constraints of the agro-food process (Gargouri et al. 2001) (Gargouri et al. 2002) such as the products pre-emption.

Indeed, the cleaning operations have a direct impact on the products quality and induce, at the same time, frequent unproductive times (Gargouri et al. 2001). That is why it is necessary to minimize the total cleaning operations duration defined by:

$$DNT = \max\{0, \sum_{i \in I} DN_{ig}\}$$

$DN_{ig}$ is the cleaning operation duration between finished products $i$ and $g$ such as is ordered just front $g$ in the current solution. Let us note that $DN_{ig} \neq DN_{ig}$.

On the other hand, possible delivery delays introduce considerable loss for the agro-food industry (lapses finished products, dissatisfaction of the customers,...). So, it is imperative to minimize the total delivery delay of tours $RT$ defined by:

$$RT = \max\{0, \sum_{k=1}^{m} R_{liv_k}\}$$

with $R_{liv_k}$ is the delay engendered for a tour $T_k$ defined by:

$$R_{liv_k} = \max\{0, \left(\max_{i \in I} d_{ijk} - \left(d_{liv_k} - (DT_k - R_k)\right)\right)\}$$

Let us note that this definition takes into account:

- the production end date of the tour commands $\max_{i \in I} d_{ijk}$, $i \in I, j \in J_{ak}$
- and, the wished production end date or still the dispatch date $d_{liv_k} - (DT_k - R_k)$ which takes into account the dynamic lorry delay $R_k$.

So global problem consists in:

$$\min imize(\alpha DNT + \beta RT)$$

with $\alpha$ and $\beta$ the respective level-headedness parameters of DNT and RT constraints.

By considering only the objective of minimization of the cleaning operation duration, problem can come down to the Travelling Salesman Problem (TSP) which is a classic combinatorial optimisation problem known to be NP-hard (Potvin 1996).

By considering of more the problem of synchronization of the production with the distribution, the reserved scheduling problem becomes even more complex.
Many exact and heuristic algorithms have been developed to solve this problem (Potvin 1996) (Lévy 1994), but an evolutionist algorithm can be the best adapted to this problem.

**EVOLUTIONIST ALGORITHM FOR AGRO-FOOD SCHEDULING PROBLEM**

In this section, we describe the resolution approach based on evolutionist algorithms.

**Coding by Variable Sizes Groups (CVSG):**

We propose, in our work, an algorithm of creation of the CVSG which optimises the cleaning operation criterion. The purpose of this algorithm is to define, at first, from the set $\text{Ord}_i = \{O_p(\{Q_i\}_i, i \in [1,l])\}$, all the possible CVSG. Every group of the CVSG is an ordered sequence of operations having no cleaning operations between them ($\text{DN}_{\text{ck}} = 0$). The operations order of the same sequence is total because $\text{DN}_{\text{ck}} \neq \text{DN}_{\text{ck}}$. The set of these groups constitutes the set of possible genes of the individuals (or chromosomes).

The CTVG algorithm defines, in a second time, the set of the possible coding or schema. A schema is a gene combination having no operations in common. An acceptable schema has to represent the set of all operations to be realized.

**Initial Population:**

To obtain a diversified population, we propose, in this work, proceeding in the following way:

- To associate, to every coding plan, one population. The individuals or chromosomes of each population are the set of possible gene’s combinations.

- To regroup the set of obtained populations, to form the initial population.

The population obtained by this algorithm will have different schemas.

**Fitness Evaluation:**

The fitness evaluation is obtained by the following algorithm:

- **Begin** $O_p = O_p$, $RT = 0$, $\text{DNT} = 0$
- **While** $(O_p < l)$ do
  - To calculate $RT = \text{MAX} (0, RT)$
  - To calculate $\text{DNT} = \text{DNT} + \text{DN}_{i+1}$
- **End do**
- Fitness $= \alpha \text{DNT} + \beta RT$
- **End**

**Selection Probability:**

The parent chromosomes are selected for mating via proportional selection, also known as “roulette wheel selection” (Potvin 1996) (Mesghouni 1999).

**One-point Controlled Crossover:**

The one-point crossover operator is applied among two parent chromosomes from the same family (having the same schema). A random position between 1 and $L - 1$ is chosen along the two chromosomes, where $L$ is the chromosome’s length. Then, the chromosomes are cut at the selected position, and their end parts are exchanged to create two offspring. After that, we have to verify the eligibility of obtained chromosomes. Indeed, the obtained chromosomes must have the same schema as their parent chromosomes.

**Controlled Mutation Operators:**

We propose, in this paper, four mutation operators:

- Mutation by permutation: consist in switching around two operation groups selected in a random way.

- Mutation by shifting: consist in moving one or several operation groups selected in a random way.

- Mutation by dissociation: consist in dissociating a random operation of its group then in integrating it into another group of the chromosome, if it is possible.
If the other groups can not integrate dissociated operation, it will form a new group.

- Mutation by fragmentation: consist in dissociating a random quantity \( q_{ik} \) of a random operation. Afterward, the mutation operator looks for the operation group which can reinstate dissociated quantity so as to minimize the cleaning operations. If this group does not exist, separate quantity will form a group and it will be reintroduced in a random position.

SIMULATION RESULTS

We consider the following example: It is a question of 10 yoghurt varieties (vanilla, banana, ...). The following table summarizes commanded quantities, delivery dates and average duration of each tour:

<table>
<thead>
<tr>
<th>Orders</th>
<th>Tour 1</th>
<th>Tour 2</th>
<th>Tour 3</th>
<th>Tour 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>P₁</td>
<td>100 200 200</td>
<td>0 0 0</td>
<td>0 0 0 150</td>
<td>300 100 0 100 0</td>
</tr>
<tr>
<td>P₂</td>
<td>200 0 200</td>
<td>0 0 0</td>
<td>0 0 0</td>
<td>200 200 200 0</td>
</tr>
<tr>
<td>P₃</td>
<td>0 0 0</td>
<td>150 150 100</td>
<td>100 250</td>
<td>250 0 200 0</td>
</tr>
<tr>
<td>P₄</td>
<td>0 0 0</td>
<td>200 200 250 0</td>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>P₅</td>
<td>0 0 0</td>
<td>200 100 100</td>
<td>100 200</td>
<td>0 200 0 0 0</td>
</tr>
<tr>
<td>P₆</td>
<td>200 200 200</td>
<td>0 0 0</td>
<td>0 200 0 0 0</td>
<td>0 0</td>
</tr>
<tr>
<td>P₇</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0</td>
<td>200 0 0 0</td>
<td></td>
</tr>
<tr>
<td>P₈</td>
<td>0 0 0 0 200 0</td>
<td>100 100</td>
<td>250 250 150 150</td>
<td></td>
</tr>
<tr>
<td>P₉</td>
<td>0 0 0 0 200 200</td>
<td>0 0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>P₁₀</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0</td>
<td>300 200 300</td>
<td>200 200</td>
</tr>
</tbody>
</table>

Delivery date: 10/05 – 13:15, 11/05 – 12:00, 11/05 – 17:30, 12/05 – 10:25

Duration (min): 45, 30, 50, 25

The estimated delays of the Lorries fleet, associated to every tour, are presented in the following table:

<table>
<thead>
<tr>
<th>Lorry delay (min)</th>
<th>Tour 1</th>
<th>Tour 2</th>
<th>Tour 3</th>
<th>Tour 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>35</td>
<td>60</td>
<td>45</td>
<td>20</td>
</tr>
</tbody>
</table>

The following table summarizes the durations of the cleaning operations between various varieties:

<table>
<thead>
<tr>
<th>P₁</th>
<th>P₂</th>
<th>P₃</th>
<th>P₄</th>
<th>P₅</th>
<th>P₆</th>
<th>P₇</th>
<th>P₈</th>
<th>P₉</th>
<th>P₁₀</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>10</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>10</td>
<td>20</td>
<td>30</td>
<td>0</td>
<td>50</td>
<td>40</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>0</td>
<td>20</td>
<td>30</td>
<td>20</td>
<td>40</td>
<td>20</td>
<td>10</td>
<td>50</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>10</td>
<td>0</td>
<td>30</td>
<td>20</td>
<td>30</td>
<td>10</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>30</td>
<td>50</td>
<td>20</td>
<td>40</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>40</td>
<td>0</td>
<td>20</td>
<td>0</td>
<td>30</td>
<td>20</td>
<td>50</td>
<td>40</td>
</tr>
<tr>
<td>50</td>
<td>50</td>
<td>0</td>
<td>20</td>
<td>30</td>
<td>20</td>
<td>0</td>
<td>40</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>20</td>
<td>30</td>
<td>20</td>
<td>20</td>
<td>40</td>
<td>50</td>
<td>50</td>
<td>0</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>50</td>
<td>40</td>
<td>40</td>
<td>50</td>
<td>30</td>
<td>20</td>
<td>20</td>
<td>30</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>50</td>
<td>50</td>
<td>20</td>
<td>30</td>
<td>60</td>
<td>40</td>
<td>20</td>
<td>50</td>
<td>40</td>
<td>0</td>
</tr>
</tbody>
</table>

In the first resolution phase and from the previous table, we have constituted the 9 following schemas of CSG:

| Schema 1 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₁₀ P₉ P₅ P₆ P₇ P₈ P₉ |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Schema 2 | P₂ P₅ P₆ P₇ P₈ P₉ P₁₀ P₁ P₃ P₄ P₅ P₆ P₇ P₈ P₉ |
| Schema 3 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₉ P₁₀ P₃ P₄ P₅ P₆ P₇ |
| Schema 4 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₉ P₁₀ P₁ P₃ P₄ P₅ P₆ |
| Schema 5 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₉ P₁₀ P₁ P₃ P₄ P₅ P₆ |
| Schema 6 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₉ P₁₀ P₁ P₃ P₄ P₅ P₆ |
| Schema 7 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₉ P₁₀ P₁ P₃ P₄ P₅ P₆ |
| Schema 8 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₉ P₁₀ P₁ P₃ P₄ P₅ P₆ |
| Schema 9 | P₁ P₂ P₃ P₄ P₅ P₆ P₇ P₈ P₉ P₁₀ P₁ P₃ P₄ P₅ P₆ |

The 9 families, constituted from previous schemas, are initially characterized by the following Fitness averages:

<table>
<thead>
<tr>
<th>Families</th>
<th>Fitness average of global criterion (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4769</td>
</tr>
<tr>
<td>2</td>
<td>4793</td>
</tr>
<tr>
<td>3</td>
<td>4802</td>
</tr>
<tr>
<td>4</td>
<td>4807</td>
</tr>
<tr>
<td>5</td>
<td>4800</td>
</tr>
<tr>
<td>6</td>
<td>4809</td>
</tr>
<tr>
<td>7</td>
<td>4800</td>
</tr>
<tr>
<td>8</td>
<td>4801</td>
</tr>
<tr>
<td>9</td>
<td>4809</td>
</tr>
</tbody>
</table>

To simulate the evolutionist algorithm, we chose \( P_m = 0.07 \) as mutation probability, \( P_c = 0.9 \) as crossover rate and \( N_{MAX} = 200 \) as a maximum number of generation.

The simulations purpose being the study of the global criterion convergence, we propose an example of convergence graphs for 3 families (1, 5 and 9):
Through the simulations of the various families, we tried to explain the interest of coding by variable sizes groups (CVSG).

Indeed, this choice allows giving solutions having different schemas but the same average performance of the global criterion, since all families converge on the same mean value of the global criterion, according to the precedent graphs.

The best solution is among those of the set best solutions relative to every family.

**CONCLUSION**

In this paper, we have proposed an approach based on evolutionist algorithms to resolve the scheduling problem in agro-food industry seen as a synchronization problem between the production and the distribution.

The obtained results are very encouraging and promising. Indeed, they show the advantages of the chosen decoding process which gives a set of different better solutions but of equal average performance.

We aim later:

- To improve the controlled crossover and mutation operators of our evolutionist algorithm.
- To introduce new optimisation criteria better to react to the chances of production and distribution
- To generalize method to form a global system integrating the global supply chain.

**REFERENCES**


RELIABILITY OF LARGE SCALE CONVEYOR SYSTEMS
A Simulation Approach

Jaap A. Ottjes and Gabriel Lodewijks
Delft University of Technology, Faculty of
Mechanical Engineering and Marine Technology
Mekelweg 2, 2628 CD Delft, The Netherlands
J.A.Ottjes@wbmt.tudelft.nl

KEYWORDS
Transport, conveyor system, reliability, process oriented simulation

ABSTRACT
The overall performance of industrial conveyor systems for bulk depends on the reliability of individual system components and their arrangement. A model for dimensioning of large scale systems in the early design stage is discussed. In the model discrete event simulation is used for the modelling of equipment reliability and simple numerical integration is applied for the stacking and loading processes. With the model stock pile dimensions, production flow and service rates of a mineral export terminal can be determined. The use of the model is illustrated with a case. The model is generic with respect to equipment break down distributions and customer characteristics. In case of different equipment arrangements the model logic has to be adapted.

INTRODUCTION
Conveyor systems in the mining industry consist of serial and parallel connections of equipment. Equipment may be mining equipment, belt conveyors, train load-out stations, stacking and reclaiming equipment and (ship) loading equipment. Some stages in the transportation chain are uncoupled by a stockpile or redundantly implemented. Besides redundancy, another reason for decoupling conveyor system often is the difference in system dynamics (Lodewijks, 2001). The overall reliability is composed of the individual reliabilities of all equipment. Reliability is defined as the average percentage of time the material flow is guaranteed. System reliability determines the service rate that can be offered to customers. Customers in this case are buyers of the minerals quarried in the mine. They collect the product by train or ship. In some cases the customer is a power plant that is directly connected to the mine.

In this paper we focus on sea transportation and define customer service in terms of waiting time of ships that collect the product. Apart from the characteristics of the loading system, the arrival pattern of ships plays an important role as well. Statistical analysis of such a system is intractable. Simulation offers a flexible way of analysing complex systems. A model will be presented and a case will be worked out to illustrate the use of the model. The model has been used in practice already for design and tender purposes. The case itself and the data used are fictitious. In the case the focus is on the determination of the stockpile size and on the effect of the system’s reliability on ship waiting times.

CASE DESCRIPTION
An inland coal mine is connected to a ship loading facility by means of a conveyor system. Figure 1 shows a schematic drawing of the terminal system. The material flow can be directed either to a stockpile via a stacker system or can be bypassed for direct loading into a ship. Stacking is done if there is no ship available or if the downstream line is blocked by disturbances of the downstream equipment. A

For the modelling the process-interaction approach is used (Zeigler 2000; Fishmann 2001). In this approach the system is virtually broken down into relevant element classes each with their typical attributes. By doing so an object oriented data structure of the system is obtained. Next, for all active element classes the process descriptions are to be defined. A process describes the functioning of an active element as a function of time. In this way the dynamic functioning of each element of the class is obtained. The last step is to create all necessary elements according to their classes and to start the element-processes. In the simulation model all active elements act parallel in time, synchronized by the sequencing mechanism of the simulation software.

The modelling of the equipment work-down cycle, the ship process, the stock pile and the actual conveying processes will be explained successively in pseudo code (Ottjes and Veeke 2003). Pseudo code has the advantage that it is communicable with (conveyor system) experts and can easily be implemented in formal computer code provided a proper simulation tool is used. (Healy and Kilgore, 1997; Veeke and Ottjes, 2000) A crucial conception in process interaction modelling is the ‘Advance’ clause. If it is encountered in a process then the element is halted for a certain time. During that time other elements may proceed. Consequently the processes are running in parallel.

**Equipment Work-Down Cycle**

The equipment work-down cycle time $t_c$ is defined as

$$ t_c = t_w + t_d $$  \hspace{1cm} (days)

in which $t_w$ depicts the average undisturbed working time and $t_d$ the average down time in one cycle. It is assumed that down time occurrences of the equipment are independent within the system boundaries.

<table>
<thead>
<tr>
<th>Table 1. Element Class Equipment: attributes and process</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributes:</td>
</tr>
<tr>
<td>- Name</td>
</tr>
<tr>
<td>- WorkTime_distribution // with average $t_w$</td>
</tr>
<tr>
<td>- DownTime_distribution // with average $t_d$</td>
</tr>
<tr>
<td>- Availability // $a = 100 \frac{t_w}{t_d} %$</td>
</tr>
<tr>
<td>- Standard_Flow (ton/hour)</td>
</tr>
<tr>
<td>- Actual_Flow (ton/hour)</td>
</tr>
<tr>
<td>- Working (1: operational or 0: not operational)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeat</td>
</tr>
<tr>
<td>ActualFlow = StandardFlow</td>
</tr>
<tr>
<td>Working = 1</td>
</tr>
<tr>
<td>Advance (Sample(WorkTimeDistribution))</td>
</tr>
<tr>
<td>ActualFlow = 0</td>
</tr>
<tr>
<td>Working = 0</td>
</tr>
<tr>
<td>Advance(Sample(DownTimDistribution))</td>
</tr>
</tbody>
</table>

reclaimer may add its reclaim flow to the load flow as long as there is a ship to be loaded, as long as the stockpile is not empty, and as long as the combined flow does not exceed the capacity of the ship loader. The load flow is further directed via a jetty conveyor to a ship loader. Figures 2 and 3 show examples of installed equipment.

MODELING
That implies that for the modelling one element class suffices for all equipment. The element class is called "equipment". It owns attributes and a process description as shown in table 1. The distributions used are model-input and may be of any analytical or tabulated form. Varying the distribution type or its parameters allows investigation of the influence of this variation on the outcome of the simulation.

Table 2 shows a fictitious model input with definitions of the distributions of working periods of each system component. For the stacker for example, it means that the undisturbed working periods are normally distributed with an average of 20 days and a standard deviation of 5 days. After a working period a down period follows. The average length of this period \( t_d \) depends on the availability and the average undisturbed working period. The relationship is given in the next equation.

\[
t_d = t_a \left( \frac{100}{a} - 1 \right) \quad \text{(days)}
\]

With \( a \) representing the availability. Each distribution realized during a simulation run is reproducible.

### Table 2. typical model input for capacity and availability of system components

<table>
<thead>
<tr>
<th>Eq. Name</th>
<th>Std Flow T/h</th>
<th>Availability %</th>
<th>Work Time Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overland Conv</td>
<td>1200</td>
<td>97</td>
<td>Exponential(14)</td>
</tr>
<tr>
<td>StockPile Conv</td>
<td>1200</td>
<td>97</td>
<td>Exponential(21)</td>
</tr>
<tr>
<td>Jetty Conv</td>
<td>3600</td>
<td>97</td>
<td>Exponential(21)</td>
</tr>
<tr>
<td>Loader</td>
<td>3600</td>
<td>97</td>
<td>Uniform(15)</td>
</tr>
<tr>
<td>Stacker</td>
<td>1200</td>
<td>90</td>
<td>Normal(20,5)</td>
</tr>
<tr>
<td>Reclaimer</td>
<td>2400</td>
<td>85</td>
<td>Table(2,0,6,7,0,45,100)</td>
</tr>
</tbody>
</table>

### Ship Modelling

A mining production facility usually has customers that collect approximately fixed loads of product by ship on a regular basis. Their ships arrive at regular times and the demand is a constant. Customers may originate from all over the world.

Ships are supposed to be characteristic for their origin. In the model for each origin a characteristic ship is created that moves the total year demand to its origin. It has a fixed load demand and a cycle time that corresponds with the yearly total demand. The cycle time may have a random component. The ship class is defined in table 3. Statistics of the ShipQueue and the JettyQueue are monitored during simulation.

In table 4 the input file used in the model is shown.

### Table 3. Element Class ship: Attributes and Process

<table>
<thead>
<tr>
<th>attributes:</th>
</tr>
</thead>
<tbody>
<tr>
<td>-DWT</td>
</tr>
<tr>
<td>-TonsLoaded</td>
</tr>
<tr>
<td>-Year Demand</td>
</tr>
<tr>
<td>-CycleTime distribution // average and deviation</td>
</tr>
<tr>
<td>-Tug handling data</td>
</tr>
<tr>
<td>-Preparation data</td>
</tr>
<tr>
<td>-Hatch handling data</td>
</tr>
</tbody>
</table>

### Table 4. Typical Ship Input

<table>
<thead>
<tr>
<th>(1) Destination</th>
<th>(2) DWT (Tons)</th>
<th>(3) Year demand (kTons)</th>
<th>(4) Std.Deviation ship cycles time (%)</th>
<th>(5) Tug to Jetty time (h)</th>
<th>(6) Time prior (h)</th>
<th>(7) Hatch Time (h)</th>
<th>(8) Time after (h)</th>
<th>(9) Tug from Jetty Time (h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>'Europe'</td>
<td>155 1950 20 165 355 549 520 50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Germany'</td>
<td>110 650 20 145 310 311 440 40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Taiwan'</td>
<td>110 650 20 145 310 311 440 40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Italy'</td>
<td>70 650 20 115 370 249 480 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Hong Kong'</td>
<td>70 390 20 115 370 249 480 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Indonesia'</td>
<td>70 390 15 115 370 249 480 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Korea'</td>
<td>55 390 15 110 275 208 370 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Hawaii'</td>
<td>55 650 20 110 275 208 370 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Malaysia'</td>
<td>55 195 20 110 275 208 370 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Japan'</td>
<td>40 1170 15 100 325 176 395 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'USA'</td>
<td>30 1300 20 55 240 136 245 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'Australia'</td>
<td>40 715 20 155 365 126 515 50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### The Stockpile

The stockpile class is defined in table 5. Its process is monitoring its contents at regular times and registering it for further statistical analysis.
The Actual Conveying Processes

Until now only the equipment work-down cycle is modelled. From that we know for each machine its status in terms of possible and actual material flow. In order to get an accurate representation of the conveyed material flow and the ship loading operation, three system components are relevant: The Stacker, filling the stockpile, the Reclaimer taking the material from the stockpile and the Shiploader that loads ships one by one. The production (tons) of each can be determined by integrating the relevant material flows. In the simulation model the material transportation time is ignored, which means that the conveyor system’s dynamics in terms of mass are ignored. (Lodewijks 2002). As a consequence of ignoring material transportation time, material produced by the Reclaimer is loaded by the Shiploader into the ship instantly. Therefore the Shiploader and Reclaimer can be combined. We solve the stacking and loading process by simple numeric integration incorporated in the processes of stacker and Loader/Reclaimer.

Table 6. Element Class Stacker: Attributes and Process

<table>
<thead>
<tr>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>- StackBatch</td>
</tr>
<tr>
<td>- TimeStep // 0.5 hours is used</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeat</td>
</tr>
<tr>
<td>- Advance TimeStep</td>
</tr>
<tr>
<td>- StackBatch = TimeStep * Stacker.Working * OverlandConveyor.ActualFlow</td>
</tr>
<tr>
<td>- if StockPile.Contents &gt; MinContents then StockPile.NotEmpty = 1</td>
</tr>
</tbody>
</table>

Mind that both the ShipLoader and the Reclaimer have their own break down processes. The Stacker element class is shown in table 6. The Reclaimer/ShipLoader element class will further be referred to as “Loader”. It is described in table 7. In the loader both reclaiming and ship loading processes are incorporated. The loadBatch may either be formed by the direct flow from the mine or by the reclaimed flow from the StockPile or by the sum of both.

Table 7. Element Class Loader: Attributes and Process

<table>
<thead>
<tr>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>- StockBatch</td>
</tr>
<tr>
<td>- LoadBatch</td>
</tr>
<tr>
<td>- DownStreamWorking (0 or 1)</td>
</tr>
<tr>
<td>- TimeStep // 2 hours is used</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeat</td>
</tr>
<tr>
<td>- Advance TimeStep</td>
</tr>
<tr>
<td>- if LoadShip = Nil then</td>
</tr>
<tr>
<td>- LoadShip.TonsLoaded = LoadShip.TonsLoaded + LoadBatch</td>
</tr>
<tr>
<td>- if StockPile.Contents &lt; StockPile.MinContents then StockPile.NotEmpty = 0</td>
</tr>
<tr>
<td>- if LoadShip.TonsLoaded &gt; LoadShip.DWT then</td>
</tr>
<tr>
<td>- LoadShip.Resume</td>
</tr>
<tr>
<td>- LoadShip = Nil</td>
</tr>
<tr>
<td>else LoadBatch = 0</td>
</tr>
</tbody>
</table>

DESIGNING

In this section the global design process using the model will be discussed. The role of equipment reliability is essential in this example. We suppose that the availability is only approximately known because the system is intended to function in a hostile climatological environment with a lack of skilled maintenance personnel. In a real case the design process would start with an expert estimate of the availability and operation of each piece of equipment.

We use the availability data of table 2 and for all equipment a negative exponential distribution with average 21 days for the working time period. The down time is taken to be uniformly distributed between 0.5 \( t_d \) and 1.5 \( t_d \). Further we use table 4 for the ships input data. This configuration is called “reference case”. Planned overhaul is not included in the analyses. Improving maintenance however may be a means to improve system performance.

Prior to the system analysis the standard run length is determined. To that end a number of runs have been performed with the reference input but different random streams for the distributions used. It appears that with a run length of 10 years the standard deviation with
respect to the average ship waiting times amounts to 5%. The standard run length is set to 10 years.

Entries 1576 90% Quantile 100.3
Mean 36.5 Std. Deviation 45.1

Figure 4a: Distribution of Ship Waiting Times. Reference Run with Stock Pile Capacity of 350 kTon

Entries 8759 90% Quantile 283.3
Mean 177.7 Std. Deviation 82.2

Figure 4b: Distribution of Stock Pile Contents. Reference Run with Stock Pile Capacity of 350 kTon

The individual runs with the model provide distributions of ship waiting times and stock pile contents. Figures 4a and 4b show examples of that. These statistical data are used to further carry out the analysis. First the order of magnitude of the required stockpile capacity is to be determined. This is done by measuring the ship-waiting times under variation of the stockpile capacity. The results are shown in figure 5. Both the average ship-waiting times are plotted and the 90% percentiles of the waiting times.

A certain point on the 90% percentile curve means that 90% of the waiting times were lower than the corresponding waiting time value. From this graph it was concluded that a ‘safe’ stockpile capacity would be 700,000 Ton. Next the sensitivity of that choice for variations in the equipment availability is evaluated. By varying the equipment availability around the reference values in table 2 in steps of 1%, figure 6 is obtained. It shows the percentage of time the stockpile has been empty and consequently could not function well. It appears that the reference availability values have some margin of 1% left.

Finally the influence of availability variations on the ships waiting times is determined. The results are given in figure 7 showing that the average ship waiting time for the reference case will be about 36 hours. A demand on the customer service may be that the average waiting time of ship before loading should not exceed 36 hours. If this demand for example would be 24 hours, a way to reduce the waiting times down to 24 hours is to improve the availability by improving maintenance, or to increase the capacity of the system components. Several other ways to find methods for tuning the system and improving the overall performance are possible but not further discussed in this paper. Some are:

- Investigating the influence of each piece of equipment separately.
- Determination of the influence of ship-arrival time deviations. One way of improvement could be the coordination of the ships-arrivals.
- Varying the shape and parameters of the down-time distributions and down time distributions that are based on measurements and experience from practice.
CONCLUSIONS

A model is presented to determine production flow and service rate of mineral export terminals taking into account the reliability of the individual system components. The use of the model is illustrated with a case. First the stock pile dimension is determined and after that the sensitivity of the ship-waiting times, for the reliability of the equipment is investigated. The model is generic with respect to equipment break down distributions and number, arrival patterns and demand of customers. For terminals with a different equipment composition the model-logic has to be adapted.

REFERENCES


ANALYSIS OF SUPPLY CHAIN DISRUPTIONS USING COMPUTER SIMULATION

Ioannis S. Papadakis
Assist. Professor of Decision Sciences
LeBow College of Business, Drexel University
3141 Chestnut Street, Philadelphia, PA 19104, USA
tel: +215-895-0225, fax: +215-895-2907, isp@drexel.edu

KEYWORDS

INTRODUCTION

Preventing supply chain disruptions and ameliorating their adverse effects is a managerial goal achievable by investments in risk management and emergency planning. Each individual entity in a supply chain benefits from disruption management investment and, due to tight interdependencies in a supply chain network, the network as a whole - as well as society in general - gains by supply chain resilience. Disruption management is, however, a challenging task involving complex calculations, multidisciplinary perspectives, and, most importantly, devastating consequences, if events are not managed effectively.

We maintain that, computer simulation, is an essential and often necessary tool for disruption risk management and emergency response planning. This fairly inexpensive information technology permits calculations to occur essentially instantly, so that representative disruption scenarios can be tested in real time. Furthermore, a multitude of operations performance criteria can be evaluated in computer models of supply chains, thus enabling multidisciplinary teams to achieve consensus about the most appropriate management policies. These two, already impressive, advantages, fast calculations and facilitation of multidisciplinary research, is complemented by a third. Computer simulation software packages are designed to unleash the imagination and creativity of decision makers so that both present and future performance can be evaluated. Therein lies its most valuable benefit for disruption management, computer simulation helps train insights and intuitions so that managers can handle disruptions with prepared minds, arguably the most important asset in emergency planning.

MODEL DESCRIPTION

Entities Comprising The Supply Chain Model

We incorporate a Make To Order (MTO) producer to a small supply chain network involving two suppliers and one collocated producer using the same production input. In the personal computer industry MTO producers are considered more vulnerable to supply disruptions and input price shocks than producers using a Make To Forecast (MTF) supply chain model (Papadakis, 2003). Can MTO supply chains increase their resilience with better risk management strategies? We build a model that can help the investigation of a long-term policy, location near to the preferred customer of a key supplier, and a policy that increases the adaptive efficiency of the supply chain, investing in capability to forecast disruption duration immediately after it occurs.

Since we study supply chain disruptions, we observe the impact of a temporary cessation in supply flow to all affected parties. We do, however, focus on the MTO producer when it comes to business continuity policies. Naturally, the effect to facilities other than the one owned by MTO is of interest when one considers various cooperative plans to manage emergencies and when one assesses long term relationships with supply chain partners. Figure 1 graphically depicts our MTO supply chain model.

A concise description of the elements of this system follows:

Producer 1 (P1) is a MTO producer facing a stochastic but stable demand. Clearly, there are many stochastic models that can be used to capture the behavior of P1’s demand. In our simulation, order interarrival times were made to follow the exponential distribution, but other probability models can be easily applied. After arrival, orders are processed
as follows. P1 mixes two inputs to produce its final product. The one input, L, is produced locally. The other input, R, is produced by a remote supplier. Input R is warehoused by the producer as it arrives in daily deliveries. In order to achieve economic plausibility, we do not provide for a warehouse of input L. Input L is warehoused by its producer, situated close to P1, and there is no duplication of warehousing investments. Clearly, in a very uncooperative mode of operation, the supply chain would include multiple warehousing facilities for product inputs L and R. And multiple facilities as well as concomitant high inventory volumes would make a business continuity problem easier to solve. It is our impression though that this very inefficient scheme is not observed in practice often.

P1’s throughput can take two levels, one corresponding to normal operation and another corresponding to disruption recovery. The former level should be enough to keep the backlog in equilibrium during normal operation at a level that minimizes costs. The latter should be higher than usual demand and provide the capability to MTO to recover fast from a prolonged disruption. Recovery production rate should be high enough to bring average backlog down to normal operation levels fast after a disruption that has caused it to increase to unacceptably high levels. Naturally, if disruption occurs while P1 operates at a very high utilization level the difference between normal production rate and recovery rate is not likely to be high. Hence, the period while the producer operates at recovery speed would be long.

Figure 2 shows the backlog level during normal operation, as well as before and after a disruption. During disruption all production units are off-line, thus backlog increases with demand. New orders keep arriving increasing the backlog up to a point when new orders are denied. After disruption end production needs to resume at recovery rate in a way that production rate is enough to both cover normal demand load and reduce average backlog to normal levels. Denying orders or rationing demand tends to have a deleterious effect to the relationship of P1 with its customer base. Hence, it is sensible to assume, orders are denied only if operating at the maximum possible level, and maximum production speed is not enough to return backlog from its maximum level to normal at a reasonable time.

**Local Supplier of Input L (SL)** produces in batches and has some flexibility with respect to batch size variation from one period to the next. SL maintains a Warehouse for input L (WL) and distributes to its customers in single units and on a just-in-time basis. There is some random variation in batch processing times for SL, but it is possible for SL to maintain a warehouse that nearly always satisfies demand for P1. This is achieved by SL’s denying its product to customers other than P1 if its inventory falls below a safety level equal to P1’a usual demand.

**Producer 2 (P2)** uses input L only, as raw material for its product. P2 operates in a different market that assigns lower value to input L than P1’s market. Thus, P2 accepts input L at less favorable delivery terms. SL delivers to P2 only if there is enough inventory reserved to satisfy P1’s demand. This unevenness in delivery terms makes P2’s production much more prone to idling and raises its production cost. Is this possible in a competitive market? The answer is affirmative, if P2 has only inferior alternative sources for input L and prefers higher production cost to lower input quality.

**Remote Supplier of Input R (SR)** delivers daily input R to P1’s warehouse. SR is assumed to have many customers, thus SR’s delivery volume flexibility is very high. SR can deliver low volumes on one day and high the next, as P1 consumes only a fraction of its total production volume. These deliveries can be handed over using various modes on transportation.

We focus on a supply disruption that can be caused by a temporary cessation in delivery on input R. Under this disruption scenario the standard mode of transportation (air and truck) halts for a period and this causes P1 to either idle its production capacity or order using a different transportation mode (surface only). Of course this second mode of transportation is slower and possibly not as reliable as air. If the disruption lasts for a long period of time, then going to a different transportation mode is a valuable business continuity option. If, however, disruption lasts for a short period of time and ends while input R is en route via the alternative mode, then the disruption is prolonged or the stopgap quantity ordered has to be written off due to limits in warehousing capacity for input R at P1’s location.

**Computer Implementation**

Our supply chain model was implemented using the ARENA 7.01 language (Kelton et al., 2003). From the point of view of supply chain disruption management, we deem that languages with Arena-like capabilities such as model animation and visual programming are very useful. Animation facilitates model verification and helps analysts to easily assess a failure mode. Visual programming permits fast modifications of the supply chain model, which becomes a platform for analysis of multiple failure modes. As a result disruption analysis gains some economies of scope. Once the basic model is validated, then most of the work needs not be repeated when more than one failure mode is analyzed. It appears that this economy of effort, is likely to make analysts more likely to gain the diverse perspectives necessary for emergency management by examining multiple failure modes. In depth analysis of only one failure mode rarely guaranties preparedness.
SIMULATION RESULTS

We investigate only two disruption modes out of the many that may be analyzed using our model. The first disruption mode examines the effect of loss of continuity in deliveries of input L due to SL’s going off-line unexpectedly (Mode A). The second disruption mode examines the effect of an unexpected cessation of deliveries of input R due to problems in transporting via the air mode (Mode B).

Both modes are very important for P1’s business continuity planning. Mode A examines the potential vulnerability for P1 if one major supplier fails to make deliveries. In addition, Mode A exhibits the effect delivery contracts and long-term location decisions have on supply chain resilience. Mode B examines the impact of loss in infrastructure necessary for commerce and supply chain operations. Reliable air freight transportation is necessary for inter-regional and international commerce and is taken for granted in developed countries. There are quite a few causes, though, for an otherwise reliable air-freight supply link to brake. From inclement weather to terrorist acts, a range of low-probability but high consequence events have resulted in the past in long and severe disruptions of supply chains. Hence, Mode B describes another supply disruption risk that is highly likely to be present in the risk portfolio of every business continuity planner.

MODE A: SL Facility Off-Line

At first we consider the case where supply is disrupted due to an interruption in the flow of input L, the locally produced input shared between producers P1 and P2. After SL’s facility goes off-line the warehouse quantities of input L begin depleting fast. P2 due to inferior terms of delivery is denied quantities of input L first. At that point production in P2’s facility halts. Soon after, inventories of input L run out and P1 also is denied delivery. Production at P1’s facility halts. P1 stops orders for input R when its warehouse becomes full. P1’s average order backlog increases linearly depending on disruption duration. After a certain point the backlog exceeds a maximum quantity at which point no new orders are accepted by P1 and the number of denied orders increases. When disruption ends, quantities of input L become immediately available to P1 and production begins at recovery speed (full capacity utilization). P1’s average backlog goes down at a speed that depends on recovery production rate. At some point average backlog reaches normal levels and production continues at normal rate.

Result A.I P2’s CAPACITY UTILIZATION IS LOWER DURING AND AFTER THE DISRUPTION. Clearly, as P2 receives quantities of input L only if P1 has enough to meet production requirements, the period of production idling is longer for P2 than for P1. Idling begins earlier for P2 and ends later. What is not as easy to recognize before simulation is that P2 suffers from production shutdowns long after disruption ends. While P1 operates at full production speed (which is always not less than normal production speed) inventories of input L are depleted faster and P2 is denied deliveries. This does not occur when P1 operates at normal production speed. Thus, P2 is exposed to capacity idling for two reasons, due to input L flow interruption (primary effect) and due to P1’s operating at faster production speed during recovery (secondary effect).

Result A.II THE LENGTH OF THE RECOVERY PERIOD TENDS TO BE INDEPENDENT OF DISRUPTION DURATION AND IT DEPENDS MAINLY ON RECOVERY SPEED. In most disruption duration scenarios analyzed, disruption lasts long enough for backlog to exceed maximum level and, hence, for orders to be denied. As a result, backlog level at disruption end is in most cases at maximum level and does not depend on disruption duration (see Figure 2). Surprisingly, the recovery period depends mainly on the difference between recovery and before-disruption production speed. Clearly, if disruption occurs when there are many orders and before-disruption production speed is close to the limits of capacity utilization, then there is little room for escalation of production speed in order to achieve a speedy recovery. The opposite is true when disruption occurs during a slow production period. Recovery duration can be minimized by P1 by raising radically production speed after the end of disruption. Backlog levels in this case go rapidly down to meet normal levels at which point production speed returns to before-disruption level. This acts as a signal to P1’s customers that response time and customer service is back to normal levels. From the point of view of the customer disruption ends at this time.

Result A.III LONG-TERM PRODUCTION STRATEGIES AFFECT PERFORMANCE DURING SUPPLY CHAIN DISRUPTIONS. Facility location is one of the most long-term decisions in operations management. It is usually dominated by considerations relating to proximity to customers, suppliers, raw material, and labor. P2 may choose a location close to a high quality supplier in an area with high concentration of related producers. This choice has many benefits and under normal operating conditions may result in high profits. If securing a much desired location results in compromises with respect to delivery terms, then performance
during disruptions may suffer as a consequence. This is P2’s story. P1 on the other hand has contracted for better delivery terms that immunize it to a certain extent against small supply chain disruptions and mitigate the impact of larger disruptions. This long term policy (favorable delivery terms) is a passive disruption management option, it does not depend on after disruption performance.

MODE B: Transportation by Air of Input R Interrupted

The second failure mode we consider permits the investigation of active disruption management policies that capitalize on emergency response preparedness. This time we consider an interruption in the flow of input R, the remotely produced and locally warehoused component of P1’s production. Regularly, input R is transported by airplane. When air transportation is disrupted, freight flow stops or another mode of transportation needs to be utilized. Surface modes of freight transportation (marine shipping, rail) are generally cheaper than air freight, but inferior in terms of delivery reliability. If P1 is in a position to accurately estimate the duration of a disruption immediately after it occurs, then this option of switching transportation modes may be exercised so that damages are controlled. In case disruption duration is estimated to be long, it is reasonable to order enough units of input R so that P1’s warehouse is filled to capacity. This way recovery may begin earlier. In the case of a short disruption duration estimate it makes sense to wait for the air mode to become operable again, thus avoiding the scenario of a delivery long after disruption ended. When the visibility of in-transit inventories is low (often the case with surface transportation) unexpected deliveries may lead to scrapping quantities of input R that exceed warehouse capacity.

We consider two scenarios. In the first, P1 has the capability to accurately predict disruption duration immediately after disruption onset, thus P1 can execute the mode switching option successfully when warranted. In the second, P1 has a poor estimate of disruption duration, the same average value but much higher variability. Poor information makes more likely the mode-switching option to fail leading to scrapping of input R units.

This disruption mode has much in common with the one previously examined, mode A. Order backlog builds up while P1 is forced to idle its production facility. This backlog increases, possibly exceeding the maximum level allowed, at which point new orders are denied. While P1 stops production SL sees inventories of input L to go up, even though P2 keeps consuming a portion of them. For prolonged disruptions SL’s production throughput needs to decrease or sales be attempted to alternative consumers of input L. After disruption end P1 operates in recovery mode at full production capacity up to the point when average backlog reaches usual levels.

Result B.1 LOW VARIABILITY IN DISRUPTION DURATION ESTIMATE LEADS TO SHORTER PERIODS OF IDLING FOR P1’S FACILITY. Disruption effectively lasts at most till delivery using the surface mode of choice. Shorter disruption duration in turn, results in shorter recovery period, as long as the two are directly related (i.e., backlog does not exceed maximum level and orders from P1’s customers are not denied). If on the other hand, there is high variability in disruption duration estimate, then the mode-switching option fails often. Surface mode deliveries arrive at the wrong time, after air mode deliveries have began anew and units of input R overflowing warehouse limits are scrapped. Clearly, good information about disruption duration requires investments prior to disruption. When good information is available on demand, emergency response strategies are easier to execute successfully. In addition, high in-transit inventory visibility, having benefits that far exceed emergency response considerations, is another capability that permits productive execution of active disruption management policies.

CONCLUSION

All the aforementioned results are very important for the design of more resilient supply chain networks through the utilization of both passive and active disruption management policies. Locating near a producer (P2) and using the same supplier (SL) as its preferred customer (P1) has a clear negative impact to the non-preferred producer. When disruptions are not as frequent P2’s position may not be as undesirable. Especially, if P2 gets SL’s high quality product at a good price. In fact, the presence of P2 facilitates supplier SL’s operations and reduces its exposure to the risk of demand interruption from its major customer P1 as in Mode B scenario. Naturally, good information available on-demand is a valuable asset to every manager. The same holds true to seemingly greater extend for disruption managers. On-demand information requires long-term investments in information collection and experienced purchasing managers interacting successfully with remote (possibly overseas) suppliers. Its value to disruption management is not easy to appraise quantitatively in a precise way. Simulation of Mode B, however, was successful in showing that appraising in a timely manner disruption duration can make a big difference in the case of an emergency.

Importantly, a major insight arising from this study is that supply chain disruptions are much facilitated by simulation analysis. In practice, many failure modes need to be considered when a company tries to build emergency response capabilities and mitigate disruption risk. Even though each mode can potentially be investigated with simple supply chain models, these models are radically different to each other. Simulation becomes a platform to test many and dissimilar failure modes. The cost of constructing a supply chain model is paid once and then any failure mode can be tested at little extra cost. This permits creativity in supply chain risk management. In addition, communicating the results of a simulated disruption is more immediate to decision makers with different disciplinary backgrounds. Even if one
has a good enough overview of the whole supply chain to assess disruption modes, explaining them to multidisciplinary teams can be very costly without simulation.

REFERENCES


APPENDIX

BIOGRAPHY

Ioannis (Yanni) Papadakis is Assist. Professor of Decision Sciences in the LeBow College of Business at Drexel University. He has extensive experience in business consulting and management science. His research focus is in Operations Risk Management with emphasis in Supply Chain Disruption Management, Infrastructure Network Maintenance, and Preparedness to Handle Emergencies. In his teaching he emphasizes the importance of mathematical decision models that are easy to interpret, implement, and are valuable to the organizations using them.
Distributed Application for Supply Chain Management Training

J. Manuel Feliz Teixeira
António E. S. Carvalho Brito
GEIN – Section of Management and Industrial Engineering
Faculty of Engineering of the University of Porto,
PORTUGAL
emails: feliz@fe.up.pt; acbrito@fe.up.pt

Richard Saw
Director of Executive Development of CLSCM, University of Cranfield
U. K.
email: r.j.saw@cranfield.ac.uk

KEYWORDS: Supply Chain, management games, “Beer Game”, inventory policies, “Bullwhip Effect”.

ABSTRACT

Here we present a computer application planned to be used as an interactive tool for Supply Chain Management Training. Implemented with Visual C++®, this application embeds the “Cranfield Blocks Game” (Richard Saw, 2002) network structure, and uses precisely the same demand patterns as the manual version of the game. Anyhow, instead of reducing the play to 12 reorder cycles, as the manual version does, this application extends those patterns throughout the time till any number of reorder cycles, what let the results become more useful and interesting even for didactic purposes. At the same time, the present application substitutes the classroom table by the computer screen, and can be made to run in AUTOPLAY mode, meaning the game can also be played with only one player or even automatically, with no players at all. In a certain way, this comes closer to some kind of Distributed Supply Chain Simulation, apart from the fact each lead time is fixed, as the “Cranfield Blocks Game” states, and the AUTOPLAY stock policy is empirical.

As each application communicates with the SERVER using the TCP/IP protocol, the players can be spread by different computers and even placed at different geographic locations if connected to the INTERNET.

In the end of this paper, results achieved with an automatic running session and with the involvement of a group of students from the Escola de Gestão do Porto (EGP) will be presented and compared.

1. Introduction

Interactive Games are nowadays being used as powerful tools to exercise the ability of people to manage a wide range of complex systems, from flight simulation to the control of the flow of water in dams under extreme conditions of raining, for example. Following such a tendency, Supply Chain Management also started to apply those techniques to improve the ability of the managers to resolve faster and more efficiently some problems they are expected to face in the real world. Training before to fit later is the philosophy of such an approach.

Related to the sciences of management one can already find games like the International Logistics Management Game® (ILMG), the Corporation: A Global Business Simulation, the BEEFEATER RESTAURANTS MICROWORLD®, as well as the GOLDRATT’S GAME® and many others, but it seems one of the oldest and most popular directed to Supply Chain Management is the “Beer Game”, conceived at MIT in 1960 (Sterman1989), with which the Bullwhip effect (amplification of demand at the upper levels of a Supply Chain) have been observed, as well as the importance of communication between partners stated. Many of those games, however, are still being widely used as manual tools. The “Beer Game”, for instance, can still be bought inside a box in which one will find written instructions, some other pieces of paper and some more little blocks representing the material units used for stocking. Nevertheless, at the moment it is also possible to play the “Beer Game” electronically, after David Simchi-Levi® and Phil Kaminsky® developed their Web Based Beer Game®.

In fact, there are some inconvenient features on presenting management games in traditional manual form, being the most significant: (1) the space required to play the game; (2) the need of an extremely well organized manual communication scheme between partners, by which time synchronization will be ensured (usual the job of another person); (3) the usage of paper cards to communicate demands; (4) and the non existence of a common support where the overall supply chain behavior can be easily shown and seen by all the members, during the game or at its end. The record of stock data and the computation of other decision variables are also time consuming in the manual version, and can be highly improved using a computer application. The present application, focused on the the “Cranfield Blocks Game”, and named “SCGame(cranfield)”, has as the main objective to give the players such advantages.

Once it uses a CLIENT-SERVER technology, this application has been divided in the applications “SCGame(cranfield) server” and “SCGame(cranfield) client”. We will present each of them in this paper.

2. The “Cranfield Blocks Game”

The original “Cranfield Blocks Game” uses a multi-level structure to represent the supply chain, thus being a more general game than its counterpart “Beer Game”, which usually looks at the supply chain as a linear structure. Nevertheless, the primary idea of the “Beer Game” is maintained, and one could say the “Cranfield Blocks Game” can also be seen as a more complex version of the former.

The structure of the “Cranfield Blocks Game” (Fig. 1) is based on four depots (retailers) that are connected to two central warehouses and these warehouses connected to a single factory, which in turn is served by an ideal supplier. The intent of this configuration is to let the players have a better sense of some of
the behaviour that stands for real supply chains, at least concerning the need to fulfil the demand of more than just one customer in each facility, and how such a need can already introduce a significant variability in the demand, turning more difficult to establish the optimal management policy, even when only one product is considered. That is the case of the game. As well, this configuration lets the results give an idea on how the response of the chain depends on the demand amplitudes, as different demand amplitudes are previously “established” at the last customers, directly connected to the depots.

These patterns will be maintained in the computer version precisely till the 12th cycle, but then they will be replaced by patterns generated based on random normal distributed numbers. This aspect was strictly preserved, for that results achieved with the computer version could be compared with the results obtained with the manual version.

Finally, in order to be able to compare the management of different facilities, there was introduced a classification in the computer version that probably can diverge from the criteria used by the father of this game. For our own purpose we decided to classify the management of each facility by the following criteria:

\[
\text{CLASSIF} = 100 \times \left(\frac{\text{totalDemand} - 1.5 \times \text{totalStockouts}}{\text{totalArrived}}\right)
\]

We refer to \text{totalDemand} represents the amount of demand at the end of the game, \text{totalStockouts} represents the amount of stockouts, and \text{totalArrived} represents the amount of material arrived from the supplier. Thus, this classification pretends to give an idea on the efficiency of the management, assigning a negative weight factor of 1.5 to the stockouts.

3. The CLIENT-SERVER application

To implement this game we have developed a CLIENT-SERVER application communicating by TCP/IP, being the SERVER the responsible for the customers demand generation, the initialization of the game, the synchronization and control of time and the communications between partners, and the CLIENT being the interface each manager has to play and manage the stock resources of his facility.

Before the beginning of the game, the SERVER (Fig. 3) must be launched in the Operating System (WINDOWs) for that it will wait for the clients to ask permission to connect. As each client asks this permission, the SERVER will assign to it a free facility, and will refuse any connections once the supply chain is complete. In that moment, the SERVER will then send a message \text{msg: play} to each CLIENT, to signal the game can be made to start. The time will only advance to the next cycle whenever all the CLIENTS have played the present cycle, thus ensuring the synchronism.
between them is maintained.

The SERVER is the centralised controller of the game. It not only establishes the access to the players and ensures the correct time holds for all of them, but also it controls the flow of demand between partners. This is achieved by the simple exchange of messages between the SERVER and the CLIENTS.

As it can also be inferred from the previous image, this version can be made to run for longer periods of time than the original 12 cycles, as well as it allows the initial stocks of the facilities, their Base Quantities (BQ) and the customers’ demand to be modified, even if actually they will start with the values of the manual version. Finally, this computer version gives the manager the possibility to introduce “random steps” of demand at each customer’s demand site, what probably can be used to train reactions to high instable markets, or even to make experiments concerning the flexibility or the agility concepts.

As the game runs, it will also be possible to make an idea of the overall supply chain behaviour, once the stock of the facilities can be shown graphically to the user, as the example of figure 4.

![Fig. 4 Example of the stocks of facilities shown at the SERVER](image)

As suggested in the next figure (Fig. 5), the game can be usually played in a room with each group of students around a computer, while the SERVER runs in another computer nearby, many times with its display projected in the wall as a way to let the groups access the evolution of the supply chain data, mainly played by internet if the SERVER is previously running in a reachable address.

On the other hand, the CLIENT application is the tool with which the groups manage their facilities, as well as they communicate the demand to their suppliers and serve their customers requirements. This application, less complex than the SERVER, is the real player’s interface, and can be made to run in any location of the computer network as long as it will be able to reach the SERVER by TCP/IP.

All the CLIENTS have the configuration shown in the next figure (Fig. 6), and differ from each other by the natural amounts of stock and BQs, as well as by the facility name and the designation of their customers and suppliers. The figure shows a CLIENT assigned to the Factory.

To help the player to locate better its facility in the supply chain, this application uses an image of the network where all the facilities are represented. Also there is a window where messages coming from the SERVER are displayed, and a check button of AutoPlay which lets the application react automatically to the customers demand and decide when to order new material. In any case, the criteria used on those decisions are empirical and do not follow any of the models known in practice for stock control, as the intention was to use this application to simulate the kind of “empirical” behaviour many times used in the practice for managing small stock resources.

At the same time, the CLIENT application has two other zones very important to the player, shown in figure 6: the demand zone, where demand from customers is continuously updated and the demand to the supplier affected, and a serving zone, where the user can fulfil the requirements of material to send to the customers. These operations will be executed when pressing the PLAY button, which

![Fig. 5 Example of a simple apparatus to play the game](image)

![Fig. 6 The SCGame(cranfield)_client in this case assigned to the Factory](image)

![Fig. 7 Menu Statistics](image)
then will automatically be disabled by the application and still in such state till the next time the SERVER will enable it, meaning this facility is again allowed to play.

Another feature concerning this CLIENT application is the possibility of tracking some important parameters during the game, as it is the case of the stock, the demand, the quantity arrived form the supplier, and the stockout level, what usually helps the player on making better decisions. This data will be presented in the form of a graph that will appear next to the CLIENT application as long as the menu option Game->Statistics is chosen (Fig. 7). A graph of this type, related in the case with Depot1, is shown in the next figure (Fig. 8).

Fig. 8 Graph of Stock, Demand, Arrived and Stockout level

In addition to the visualization of these parameters, the player can use the menu option data->(m, std) of this window to automatically compute the average value and standard deviation of each of the data series represented.

4. RESULTS AND COMMENTS

As was said earlier, the results presented in this paper have been obtained in two sessions. In the first session the game was made to run automatically with all the facilities in AUTOPLAY mode, during 30 demand cycles. In the second session the game has been played by seven groups of students attending a Masters Course at the Escola de Gestão do Porto (EGP), and also run during the same 30 cycles.

In each case, data like the one showed in figure 8 was collected for each facility, and then its average values, standard deviations and accumulated values have been computed, as well as the final classification of the facility. These computed results were then recorded and handled in an EXCEL spreadsheet for the two separated sessions, in order to make the comparison between them easier. Notice that the unity of stock used here is the SKU, worldwide known as Stock Keeping Unit.

Different kind of calculations could be made based on the raw output data retrieved from the game, but for our purposes we decided to present only some results based on averages and on standard deviations. Averages will give us the sense of how each facility was in general handling the interesting parameters, while the standard deviations will lets us have an idea about the variability observed on handling those parameters. Based on such simple results we expect already to be able to conclude about the interest of the present application tool.

In figure 9, for instance, the average values of those parameters are represented, computed after the game have been played the first time by the students. It shows that even if the final classification of the management of the facilities exhibit an almost flat behaviour, it was also observed an obvious tendency on the other parameters (except the demand, of course, due to be an input) to grow from the left to the right of the supply chain network, as well as when approaching the Factory. This is precisely the tendency induced in the system by the differences in the BQ quantities, as we have stated earlier. Thus, due to the fact these results tend to be proportional to the respective BQ of the facility, is seems all groups have used the same kind of empiric reorder policy, as well as it seems they have faced the same challenges on managing the facilities.

Fig. 9 Computed results obtained on the first Students Session

These results can be compared with the data obtained with the AUTOPLAY session, which is represented in the next figure (Fig. 10), where all the facilities were using precisely the same stock control empiric policy, previously established by us.

Fig. 10 Computed results obtained with AUTOPLAY Session

Although the various parameters seem to have improved slightly in the AUTOPLAY case, as well as the level of stockouts have been reduced, the two cases show a similar tendency to follow the induced BQ differences. Maybe this would mean that to manage this system in an empiric base few differences would arise due to the “experience” and the “quality” of the manager. In fact, in the two cases de classification parameter is maintained practically constant at all the facilities.
However, when we compare the variation observed in the two sessions by means of the standard deviations charts, shown in figure 11 and figure 12, a significant difference is already noticed. It seems obvious the “experienced” manager (AUTOPLAY) would be able to handle the facilities in a much more smooth way than a beginner. In fact, in the AUTOPLAY the variability of the stockouts was drastically reduced at the warehouse 1, warehouse 2 and factory, as well as the amount of material ordered, even if not as drastically.

![Fig. 11 Standard Deviations at the first Students Session](image1)

![Fig. 12 Standard Deviations in the AUTOPLAY Session](image2)

Other measures could be achieved with this application, as it gives the user a wide range of untreated output values over time, so making the experiments of management dependent on the imagination of the game’s supervisor. Also, as this game lets the players visualize and record the evolution of important parameters with the time, even tests concerning the visibility or not-visibility between partners can be made, depending whether the overall supply chain is projected or not-projected on a wall, where all the stock levels can be turned visible to every partners.

5. CONCLUSIONS

Based more on the potential of this application than on the results presented here (these are just an illustration of such a potential), we conclude this computer application represents an interesting tool for Supply Chain Management Training. In fact, the results achieved with the manual version can also be obtained using this tool, and in addition other interesting aspects are made available, like the portability and the flexibility of a distributed computer application, the capability of considering visible or not-visible scenarios, as well as the introduction of demand steps at the ultimate customers, which we imagine can be useful in future tests concerning studies of Flexibility or Agility, for example.

References:

5. This game is described by Goldratt in his "novel", the Goal (North River Press, 1992, pp. 104-112.). It can be used to illustrate many concepts in Production Management.
6. Professor at the Massachusetts Institute of Technology (MIT)
7. Professor at the Industrial Engineering and Operations Research Depart., University of California
SIMULATION OF A SUPPLY CHAIN WITH A LOGISTICAL AND ECONOMICAL APPROACH

Alejandra Gomez-Padilla
Gestion Industrielle Logistique et Conception (GILCO) Laboratory
Ecole Nationale Supérieure de Génie Industriel
Institut National Polytechnique de Grenoble
38000, Grenoble,
France.
E-mail: gomez-padilla@gilco.inpg.fr

KEYWORDS
Industrial engineering, Optimization, Performance analysis

ABSTRACT

In this paper we present our work in process, consisting in the study of a supply chain formed by a supplier and a retailer. Our analysis is in terms of the contractual relation and the inventory control system. There is only one stocking point and it is the retailer who holds the stock. We describe the different modalities we can find in the literature. Then we explain our modelling methodology as well as some of the models with which we are working. The contracts that we consider are wholesale price, buy back and sales rebate. The stock policy we consider is base stock. We show the influence of the economical decisions of the contracts over the logistics service, specifically the inventory control policies.

CONTEXT

Our interest is in confronting different contracts with different ways to calculate the base stock level. In order to do this, we consider a supply chain consisting of two agents: a supplier (upstream company) and a retailer (downstream company). The upstream company provides a product to the downstream one. There are three kinds of flows between them: informational, financial and physical. The two companies are linked by a contract in which they specify how the financial and physical flows are going to take place. It is by the contract that the companies establish the quantity of product that the retailer is going to buy to the supplier, or in other words, the quantity the supplier has to produce and the retailer to sell. It is also by the contract that the companies stipulate how the retailer is going to pay for the product and if the supplier has to pay the retailer, as well as the way to establish on what these payments depend. The three contracts that we consider in this paper are wholesale price, buy back and sales rebate; they will be explained along with the problem formalization. The informational flow is going to give context to their relation. The retailer has to satisfy the market (external to the chain); the information it has is: the expected demand, the price of the product in the market and its own internal costs (production, holding and shortage). The supplier has to produce a certain quantity of product (which will be the physical flow), this is why it has to assure that its suppliers are going to provide the components to produce; the information it has are its own costs.

In literature we can find very different approaches to contracts and inventory control. The works on contracts concentrate mainly on studying the coordination mechanisms between the members of the supply chain for one or several types of contracts as (Larivière 1995; Cachon 2004; Tsay 1999). They analyse the incentives to force the physical flow of a quantity in one type of contract so that it is possible to maximise the benefit for one of the agents. Other works are oriented to identify different methodologies to accomplish what was previously stipulated in a contract in terms of physical flow (Bassok and Anupindi 1997; Tsay 1999).

Concerning inventory control, we find models that consider multiple retailers and multiple suppliers for one stocking point (Ganesan 1999), or multiple stocking points (Clark and Scarf 1960; Cachon and Zipkin 1999).

The two questions to which an inventory control system must answer are: 1) when to order? and 2) how much to order? (Fogarty 1991; Dupont 1998; Giard 2003). A base stock policy is a system where the periodicity to pass an order is fixed and the order quantity is variable, so the stock level has to be calculated and orders are passed to complete this level of stock. The base stock is the quantity of product that has to be always available at the beginning of each period, and it is calculated by minimising the costs, mainly the holding and shortage costs. The companies search to maximise their benefit, then, it may be possible to think that the benefit maximisation is not forcedly the same that the costs minimisation. In fact the benefit maximisation is equivalent to the costs minimisation if the holding and shortage costs are calculated according to the contract parameters. This is not always the case since usually the costs are defined without taking into account the contract, which has a real impact over the logistics service.

PROBLEM FORMALIZATION

If we consider the expected demand E(D); the expected sales of the retailer S(q); the market price of the product P; the quantity of the product that the supplier has to produce q; the expected financial flow between the agents or transfer T(q); and the supplier cost of producing one unit C, we can represent the different flows among the retailer and the
supplier as in figure 1; the internal costs for each agent are not represented. The expected benefit of the retailer is equal to the revenue made by the sold products minus the financial transfer between the agents minus the cost of stocking the product. The expected benefit of the supplier is equal to the financial transfer minus the cost for the supplier of producing the q units produced.

![Diagram](image)

Figure 1: Information, financial and physical flows within a supply chain composed of a retailer and a supplier

If we define \( \pi_r(q) \) as the expected benefit of the retailer, \( \pi_s(q) \) as the expected benefit of the supplier and \( \Pi(q) \) as the expected benefit of the chain or global benefit, considering only the external exchanges, we can represent them as:

\[
\begin{align*}
\pi_r(q) &= PS(q) - T(q) \quad (1) \\
\pi_s(q) &= T(q) - C q \quad (2) \\
\Pi &= \pi_r(q) + \pi_s(q) = P S(q) - C q \quad (3)
\end{align*}
\]

The financial flow of each one of the agents depends on the contract. Under this context the global financial flow of the chain remains constant, thus, the global benefit is independent of the financial transfer between the agents: it is independent of the type of contract. We have therefore three different functions of benefit: the one for the retailer, the one for the supplier and the one for the whole chain. If each entity looks forward on maximizing its benefit there are then three behaviours that may be in contradiction.

Attending to the financial flow, after literature review we decided to analyse three contracts: wholesale price, buy back and sales rebate. The wholesale price is a contract in which there is only one price \( w \) that the retailer is going to pay to the supplier for each unit purchased (4). In a buy back contract, the retailer pays a price \( p \) per unit purchased, and after observing the demand, the retailer pays an amount \( b \) per unsold units; this is, the supplier is not physically buying back, but rather sharing the cost of stocking units, where \( b \) is the expected stock –or the expected unsold units– (5). In the case of the sales rebate contract, the retailer buys the products at a price \( w \), and if he buys a quantity that is bigger than a fixed amount \( t \), the retailer gets a rebate of \( r \) for the units that pass this amount, if the retailer orders a quantity smaller than \( t \), the contract is similar to a wholesale price (6).

The transfer is then, respectively for each contract and for each k period:

- Wholesale price: \( T_k(q) = wq_k \) (4)
- Buy back: \( T_k(q) = wq_k - bl_k(q) \) (5)
- Sales rebate: \( T_k(q) = wq_k - q_k t \) for \( q_k \leq t \), \( T_k(q) = wt + (w-r)q_k - t \) for \( q_k > t \) (6)

The holding cost for the retailer \( h \), can be determined by 1) considering the various costs to keep a product in stock such as rent, salary of people working on the inventory, taxes, heating, etc; 2) as a percentage of the price of the product (Fogarty 1991). For the wholesale price and sales rebate contracts, the holding cost \( h \), can be calculated using either of these solutions. For the case of the buy back contract, it is easy to analytically prove that the holding cost has to also take into account the participation of the supplier. Concerning the shortage cost \( b \), it is calculated as the marginal profit that is lost when the item is not immediately available in stock (APICS 1998), this is, the difference between the market price and the price paid to the supplier for the product. For the wholesale price and the buy back contract, this cost is easy to calculate this way. For the sales rebate it is not so simple since there are two prices. Then again, if we consider our model, it is possible to analytically calculate the exact shortage cost taking into account the contract. The wholesale price contract is then the only contract for which the holding and shortage costs are the same, independently of considering the contract explicitly or not; for the buy back and sales rebate contract, it is important to consider the parameters of the contract in order to calculate the base stock that actually maximises the benefit. We are going to say from now on in this document that the base stock that minimises the costs \( (s_{\text{min}}) \) is calculated without taking into account the contract and the base stock that maximises the benefit \( (s_{\text{max}}) \) is calculated taking into account the contract.

In our work we intend to compare and visualise, by the bias of simulation, the relationship between the contract and the inventory control policies, to compare the behaviour of the contract confront the two ways of calculating the base stock level and to identify which contract is more interesting in economical terms. We seek to show how the use of the correct parameters for the base stock will maximize the benefit.

**SIMULATION**

In order to simulate, we define a number of parameters with which we calculate the base stock that minimizes the costs and the base stock that maximizes the benefit for the retailer.

In this part of our work, we simulate for a random demand following a uniform distribution between 0 and 1. The quantity to order can be between 0 and 1. We fixed the parameters as follows: the sales price on the market is \( P=10 \) and the price per unit is \( w=5 \). The buy back price for the buy back contract is \( b=0.5 \), and for the sales rebate contract the rebate is of \( r=2 \) and it will be applied to the units over \( t=0.5 \). The holding cost and shortage cost of the retailer for the cost minimisation are respectively, \( h_r=1 \) and \( b_r=5 \); for the cost maximisation they are fixed according to the contract. We will compare the results for 240 periods. The initial stock will be zero.

In Figure 2 we show the sales for the three contracts and for the two ways of calculating the base stock. As we said previously, for the wholesale price contract the two ways of calculating are equivalent. We see in this figure how the sales are higher if the base stock was calculated considering the contract in order to maximise the benefit.
Figure 2: Total sales for different contracts and base stocks

We can also visualize by simulation that, by calculating the base stock maximizing the benefit, the number of units in shortage is smaller, the average of products in stock is higher as well as the number of ordered products. In Figure 3 we show the total number of units in shortage for the three contracts and the two base stock policies.

Figure 3: Total shortage for different contracts and base stocks

In Table 1 we show the results in terms of benefit for each contract and each base stock policy. We can see that, if the retailer had to decide between the three contracts, under the proposed parameters, he would be most likely to choose the sales rebate contract.

Table 1: Benefit by Contract and by Base Stock Policy

<table>
<thead>
<tr>
<th>Contract</th>
<th>Policy</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buy back</td>
<td>Benefit max</td>
<td>535.42</td>
</tr>
<tr>
<td></td>
<td>Costs min</td>
<td>533.09</td>
</tr>
<tr>
<td>Sales Rebate</td>
<td>Benefit max</td>
<td>541.81</td>
</tr>
<tr>
<td></td>
<td>Costs min</td>
<td>541.58</td>
</tr>
<tr>
<td>Wholesale</td>
<td>Benefit max</td>
<td>491.02</td>
</tr>
<tr>
<td></td>
<td>Costs min</td>
<td>491.02</td>
</tr>
</tbody>
</table>

CONCLUSION

We have seen that the inventory control policy affects the benefit obtained by the retailer under three contracts. The results on the side of the supplier would be an interesting element to consider in future simulations. We plan to repeat the simulations for different demand distributions to see if the results follow the same tendency. In a second time, our aim is to extend this simulation protocol to other contracts in order to conclude for each contract in combination with each inventory policy.

REFERENCES


Lauriere, M.A. 2002. “Supply Chain Contracting and Coordination with Stochastic Demand”. In Quantitative Models for Supply Chain Management, S. Tayur, R. Ganesan, and M. Magazine (Eds), Kluwer, USA.


BIOGRAPHY

ALEJANDRA GOMEZ PADILLA was born in Guadalajara, Mexico and studied industrial engineering at the Instituto Tecnologico y de Estudios Superiores de Occidente (ITESO - Mexico), fishing her studies in 1998. She worked one year as process and product engineer. In 2001 she obtained a MSc degree from the Ecole Polytechnique de Montreal and since this same year she is a Ph.D. candidate on Industrial Engineering at the Institut National Polytechnique de Grenoble.
SIMULATION OF ROBOTIZED FOOD PACKAGING LINES
Hans P.M. Veeke and Gabriël Lodewijks
Delft University of Technology, Faculty of
Mechanical Engineering and Marine Technology
Mekelweg 2, 2628 CD Delft, The Netherlands
H.P.M.Veeke@wbmt.tudelft.nl

KEYWORDS
Industrial Engineering, Computer Integrated Manufacturing, Discrete Process-oriented simulation

ABSTRACT
In robotized food packaging lines the performance depends on the number of products packed per time unit, product loss percentage and the percentage of completely filled packages. This paper presents a feasibility study to investigate a new packaging concept where the product and box conveyors are uncoupled by introducing buffer positions for boxes with each packaging robot. The new concept proves to be superior with respect to all performance aspects. Above that the concept enables immediate and full-speed startup from all situations.

INTRODUCTION
In modeling food packaging, productivity questions focus on minimizing both product loss and incomplete product boxes – besides the usual maximization of production and cost minimization [PMMI, 2002] [Tompkins et al., 1996]. In this paper a new concept for the configuration of an automated packaging line is investigated and compared to a common configuration (as shown in figure 1).

Figure 1. Chocolate packaging line

The new concept shows technological advantages beforehand, but the logistic consequences were still unclear. To investigate these consequences a simulation model has been developed. In this paper the configuration and possibilities of the model are explained and the results of experiments will be described. The model aims to support a comparative study of concepts and not to optimize the configuration and control. Therefore the robot control with respect to product and box selection is kept simple for both concepts.

MODELING OF FOOD PACKAGING
A packaging configuration typically consists of two conveyor systems and a material handling system with robots. One conveyor system is being used for the transportation of products (chocolate bars, hamburgers, bread etc.), the second conveyor system transports the boxes that should be filled with products. Figure 2 shows the primary functions of the material handling system with robots.

Figure 2. Packaging functions

Two function types are distinguished:
- Control ("Plan")
- Operate ("Act")

The Plan function receives data about the state of and on the conveyor systems. The data represent the image of the products and boxes on the conveyors, their speed etc. Based on this image a selection is made and tasks sent to the robot system. For this case it is not important whether the control is centralized or not.

When we zoom in to the control function, three subfunctions are distinguished (figure 3):

Figure 3. Plan functions

- Define a subset of products: determine the set of products, which can technically be reached by a robot.
- Calculate attributes of these products that will be used for the final selection (distance, speed etc.).
- Select the product that best suites the requirements.
If a robot has picked up a product, the same functions are required to select the best box to put the product in.
The quality of the planning depends on each of the subfunctions and the conveyor information received. The model takes a communication delay of 10 ms into account. It is possible to select different selection strategies. The current model selects the product (or box) that requires a minimal robot displacement in the direction of the conveyor.
The subfunctions of a robot are shown in figure 4. First a number of product cycles is being performed. The number depends on the number of “grippers” of a robot. Each cycle consists of three steps:

- wait for (the assignment of) a product
- move to the product
- pick up the product.

Next a packaging cycle is performed, consisting of waiting for (the assignment of) a box, move to the box and “drop” the product.

Asynchronous configuration

The new configuration for food packaging lines also consists of two conveyors, but now a buffer space is present between these conveyors. Buffers are positioned at the positions of the robots. The conveyors now move in the same direction.

![Figure 6. Asynchronous Configuration](image)

MODEL INPUT

The model input can be divided in 4 groups:

1. the product conveyor
2. the box conveyor
3. the product dimensions
4. the production characteristics
5. the robot (configuration) attributes

Input data on the product conveyor, box conveyor and product dimensions are shown in figure 7. The input data of the box conveyor in the figure above concern the asynchronous configuration. The buffer size is not applicable for the synchronous situation.

Furthermore it is assumed for the synchronous situation that the speed of the box convey or is set to a value that a box can be filled exactly.

![Figure 6. Physical dimensions](image)
after passing all the robots. Suppose C is the cycle time of a robot, the number of robots is R, the number of products per box is N and the longitudinal reach of a robot is B, then the speed of the box conveyor should be:

\[(B \times R) / (N \times C)\].

The production characteristics are expressed in terms of production rate, production variance and number of products per box.

<table>
<thead>
<tr>
<th>Arrival Rate [p/min]</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. variance [%]</td>
<td>20</td>
</tr>
<tr>
<td>Products/Box</td>
<td>16</td>
</tr>
<tr>
<td>Rows</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 8. Production characteristics

The rows field is usually set to zero, which means that products can arrive at a random lateral position on the conveyor. There are situations however that products arrive in a perfect lines or rows, for example in bakeries. In these situations the variance is automatically set to zero.

Special attention is given to the generation of the correct production rate. According to figure 8 the production rate varies between 240 and 360 with an average of 300. To achieve this rate the generator in the model uses the maximum rate of 360. At arrival of a product it draws a uniformly distributed number x between 0 and 360. If x <= 300, the product is put on the belt, otherwise it is “thrown away”.

Finally the configuration and robot attributes are shown in figure 9. Robots may have more than one gripper. The model assumes that the number of products per box is an integer multiple of the number of grippers specified in order to prevent complications of picking up for example 2 products, while a box to be filled only has one empty position.

The pickup height is the distance the arm has to bridge if the gripper is positioned exactly above the product. After picking up the gripper rises 5 mm, before the robot starts the horizontal movement.
The put-down height is the vertical distance for putting down a product into a box. Effects of piling in the box are neglected. After putting down the same distance is traveled again before the robot is allowed to start a horizontal movement.

The simulation model has been developed in the discrete process simulation language TOMAS [Veeke, Otjes, 2000]. TOMAS directly connects to the informal way of describing processes as it has been followed in this paper until now [Zeigler et al. 2000].

<table>
<thead>
<tr>
<th>Robots</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
</tr>
<tr>
<td>Relative distance [mm]</td>
</tr>
<tr>
<td>Reach [mm]</td>
</tr>
<tr>
<td>Number of grippers</td>
</tr>
<tr>
<td>X-Acceleration [mm/sec²]</td>
</tr>
<tr>
<td>X-Deceleration [mm/sec²]</td>
</tr>
<tr>
<td>Y-Acceleration [mm/sec²]</td>
</tr>
<tr>
<td>Y-Deceleration [mm/sec²]</td>
</tr>
<tr>
<td>Horizontal speed [mm/sec]</td>
</tr>
<tr>
<td>Vertical speed down [mm/sec]</td>
</tr>
<tr>
<td>Vertical speed up [mm/sec]</td>
</tr>
<tr>
<td>PickUp Time [sec]</td>
</tr>
<tr>
<td>PutDown Time [sec]</td>
</tr>
<tr>
<td>PickUp Height [mm]</td>
</tr>
<tr>
<td>PutDown Height [mm]</td>
</tr>
</tbody>
</table>

Figure 9. Robot configuration attributes

TOMAS offers the usual verification support by debugging facilities. 2D Animation is being used for the verification of the control algorithm. The results of the plan functions are visualized by different colors of the products during the simulation (e.g. products within the physical reach of a robot are colored red).

EXPERIMENTS

The performance of the packaging system will be judged with 3 criteria:

- **The effectiveness of the system** is defined as the ratio between the produced number of full boxes per hour and the theoretically feasible number per hour based on the supply rate of products. For example if the supply rate is 300 products / minute and a box should contain 16 products, then a maximum of 300 * 60 / 16 = 1125 full boxes per hour can be produced. Incomplete boxes are considered production loss.
- **The percentage of product loss**; product loss means products reaching the endpoint of the conveyor without being picked up by a robot.
- **The utilization of a robot**. As a result of each experiment the average technical cycle time can be calculated; it is the time duration of one arm movement and one picking move. This duration is influenced by the position of the arriving products on the conveyor. The first robot receives the maximum supply and is able to realize a shorter cycle time than the last robot. This technical cycle time determines the maximum number of cycles per hour for a robot. The utilization is defined as the ratio between the real number of cycles and the calculated number.
During a simulation run, the first 400 seconds are considered startup time of the system. After 400 seconds the registration of data for the performance criteria starts. Each simulation run covers a clock period of one hour. Comparison of results of 10 consecutive runs did not show significant deviations with the results of one hour.

In figure 10 a screen shot of the synchronous configuration is shown. The filling degree of a box is visualized by color and brush fill of the box’s rectangle. At the bottom of the screen the current simulation time, the number of products lost so far and the number of incomplete boxes are shown, including the products and boxes production per minute.

Figure 11 shows a screen shot of the buffered configuration simulation. In this concept each robot has a buffer conveyor line at its disposal to position the boxes to be filled. In this way the box conveyor and product conveyor are uncoupled. The advantage is clear: all boxes delivered will be completely filled by definition and the box supply can be controlled independent of the product supply rate. By filling the buffer with empty boxes before production there are no startup losses. Thus is even more important when production disturbances cannot be excluded. If a robot filled up a box, it signals the plan function to provide a new empty box. The filled box shifts to the end of the buffer and is inserted between the boxes passing on the box conveyor. At arrival, an empty box enters the buffer at the first (leftmost) buffer position and is shifted as far as possible to the right by the buffer conveyor. If the buffer size is n boxes then the loading position of the robot is buffer position n – 1. The first experiments already showed that the density of boxes at the box conveyor is very low. So one extra position after the loading position is enough to guarantee a box transport in time.

Two series of experiments have been performed:
- A series with 200, 225, 250, 275, 300, 325 and 350 products/min and 16 products/box
- A series with 275 products/min and 4, 8, 12, 16 and 20 products/box.

RESULTS

The effectiveness and product loss percentages are shown below.

The effectiveness of the synchronous configuration decreases strongly with an increasing product supply rate, because the increased number of partly filled boxes. The effectiveness of the buffer configuration is completely determined by the capacity of the robots. But even when
this capacity is insufficient the system still delivers only full boxes. During the experiments the cycle time of a robot appeared to be about 0.8 sec. Therefore the maximum capacity with 4 robots is expected to be around 300 products / min. The variance of 20% causes irreducible product losses at production rates of 250 products/min or more. This is shown by the product loss percentages. Below the capacity limit of 300 products / min the performance of the buffer configuration is superior to the synchronous configuration.

With respect to the criterion of robot utilization an effect appears, which is very interesting from an organizational point of view. With the buffer configuration all idle times concentrate at the last robot(s). This means that low production rates can be easily realized with less operational robots.

With the synchronous configuration idle times appear at both the first and the last robot. Apparently, the number of operational robots required cannot be controlled with different production rates. The system shows a “wave” effect in time. The first robot has no problems with the availability of products, but may have to wait for filling space in the boxes (they are mostly filled by the other robots). The last robots has always enough empty boxes at its disposal, but experiences increasing product shortages with lower production rates.

Finally the results with respect to the influence of box sizes are also in favor of the buffer configuration. The buffer configuration is hardly influenced by the box size. The effectiveness and product loss change slightly with 4 products per box. This can simply be solved by increasing the speed of the box conveyor or by increasing the buffer size with one position. It is also possible to make the control of the boxes flow more intelligent, but this is not the subject of this research.

With the synchronous configuration the effectiveness increases with decreasing box size but it still is significantly below the effectiveness of the buffer configuration. The minimum product loss percentage is reached at 8 products / box. Below this number the product loss increases again, because the density of the boxes on the box conveyor urges the system to increase the conveyor speed.

**CONCLUSIONS**

The buffer configuration has a number of logistical advantages compared with the synchronous configuration:

- The effectiveness, measured in number of filled boxes per hour, is structurally higher.
- By definition it is impossible to deliver incomplete boxes.
- Within its technological capacity limits the percentage of product loss is negligible. The maximum measured value was 2 % with 16 products per box and 4 % with 4 products per box.
- Overcapacity manifests itself at the last robot enabling real capacity control. This can significantly increase the efficiency of the configuration.
- The configuration has no start up problems. Also after a standstill caused by a disturbance the packaging line can start at full speed immediately. In the long term this will result into an increased effectiveness.
- The speed of both the box conveyor and the product conveyor can be controlled independently. Especially for small packages the advantages of this become evident. Besides this, the number of buffer positions automatically increases with smaller box sizes.
- The configuration can be extended further than the synchronized configuration. The introduction of an extra packaging line can be postponed, because the density of the box conveyor is low.

During the simulation it was shown that the box size in the synchronous configuration is quite important. The density
of the boxes is so high that the only solution to provide enough boxes is speeding up the box conveyor; especially in situations where a small number of products per box is combined with a relative large product surface (e.g. hamburgers). The effect is however that the available time for the robots to fill a box decreases significantly. Contrary to the asynchronous configuration, which is hardly sensible for the number of products per box, the synchronous configuration has a predefined optimum number of products per box.

FUTURE RESEARCH

At this moment the configuration is being engineered and runs in prototype at some companies in the Netherlands. The logistic research now focuses on optimizing the control strategies for the Plan function. The configuration is one of the many production configurations that are subject of study in the Virtual Industrial System environment, which is currently being developed, based on the distributed simulation approach of TOMAS [Veeke and Ottjes, 2003].

REFERENCES

NEW ALGORITHMS FOR TRAJECTORY OPTIMISATION OF DISPLACEMENT OF TOOLS AND OBJECTS IN THE PRODUCTION SYSTEMS

Tomasz Szatkiewicz
Tomasz Królakowski
Błażej Balas
Wojciech Kacalak
Technical University of Koszalin
Department of Fine Mechanics
75-620 ul. Rąclawicka 15-17 Koszalin, Poland
e-mail: {szatkie, krakow}@tu.koszalin.pl

KEYWORDS
Optimisation of trajectory, travelling salesperson problem.

ABSTRACT

The issue of displacement trajectory optimisation plays a vital part in modern production systems. The determination of an optimal flow of resources or the movement trajectories of tools may contribute to a large extent to the decrease of production costs. There are numerous methods and algorithms to solve this task. They make use of various technologies: from traditional numerical methods, linear programming to innovative hybrid systems based on artificial intelligence methods. These algorithms are characterized by a varied computational complexity and accuracy of solutions. While choosing a proper algorithm one should take into consideration the number of nods on the one hand and their arrangement in the space and time available and required accuracy of the solution on the other. In this paper, the authors present a new hybrid system to solve the problem of the travelling salesman problem based on artificial neuron networks and local search algorithms.

PROBLEM ANALYSIS

The problem of displacement trajectory optimisation, depending on a particular application can be viewed as a well-known issue of travelling salesman problem (TSP) or as a problem to find the shortest way of Hamilton on a graph (Sysło 1993). In the case of the issue of a travelling salesman problem we must find the shortest way, i.e. one with the smallest cost, a closed way between all the nods in a network, one which runs through each nod once only. In the theory of graphs, this problem is identified with finding the shortest Hamilton’s cycle in the graph (Sysło 1993). When there is no need to “close” the cycle by returning the route to the starting point we deal with the problem of the shortest way of Hamilton.

![Figure 1. a) example of travelling salesperson problem, b) example of Hamilton’s way](image)

Both problems presented above belong to the so-called class of np-complete problems (Garey an Johnsoen 1979), which are characterized by the fact that time necessary to find an optimal solution ascends in an exponential manner together with the increase of the extent of the problem, that is the number of elementary displacements.

Due to an exponential computing complexity it is impossible to use methods of a complete survey, as already with the extent of the problem exceeding a dozen or so of nods the time necessary to find the solution becomes too long. Finding the solution of a task with the extent of 1,000 nods while using complete survey methods and the fastest computers available would take millions of years.

There are numerous algorithms with a complexity smaller than the exponential one and which solve the problems under discussion. Unfortunately these algorithms do not guarantee that solutions generated by them constitute a global optimum. In most cases, the problem of the selection of a proper algorithm to our task is reduced to finding a compromise between the time necessary for a given algorithm to perform calculations and the quality of the solution obtained. Algorithms of a greater computational complexity usually generate solutions which are better than faster algorithms.
METHODS OF SOLUTIONS

The travelling salesman problem has been of interest to many scientists from different fields for years now. This is so because it has many practical applications. No algorithm has been developed to solve this task in time shorter than exponential time. In recent years, a few serious attempts have been made to use artificial intelligence methods to solve the problems in question. Systems (Kacalak 1994; Szatkiewicz. 2002; Tadeusiewicz 1993) have been developed, which use Haming and Kohonen’s artificial neuron networks, flexible networks, evolutionary algorithms or simulated annealing. Each method mentioned above is characterized by different efficiency, which depends on values specific of every algorithm of coefficients.

While making a comparative analysis of the abovementioned algorithms we may attempt to compare their efficiencies. The effectiveness of a given method comprises of such factors as: the average number of steps necessary to obtain a solution that will satisfy us, its quality and a numerical complexity of making a single step of an algorithm – in the case of a neuron network of a single learning epoch, and in the case of an evolutionary algorithm of a single conversion of a population of individuals. A vital part as regards the speed of algorithms’ operation is played by the proper selection of the types of variables used, structures of data, as well as a possible selection of the proper computational unit (the processor’s architecture).

ARRANGEMENT OF NODS: ITS INFLUENCE UPON SELECTION OF ALGORITHM

The first feature to be considered of the problem to be solved is its size, that is the number of elementary displacements. This quantity has a decisive influence upon time necessary to solve a given task with the help of one of algorithms. There also exists another factor, which has a substantial influence upon the speed of the solution, namely the arrangements of nods in space, that is their typology. This factor is often rashly neglected in the assessment of the difficulty level of a given task. An extreme example may be illustrated with a problem with a large number of nods placed on a circle. Even someone who is not an expert will easily, using their intuition, solve this problem in an accurate manner, finding an optimum (see. Fig. 2a). The very same problem, due to the amount of time, cannot practically be solved with the use of an algorithm of complete survey. This is due to the fact that the arrangement of nods has a decisive influence upon the shape of the space of the states of a task being solved and it is this shape that determines the difficulty level of finding a solution of a given task. The space of states in the travelling salesman problem’s set as an example is n – dimensional, discrete and finite. The values of its lower and upper limits can be determined. In the case of a problem with nods arranged on a circle the shape of the space of states resembles one steep valley with a distinctive global optimum. In the case of a few distinctively separated concentrations of nods (see Fig. 2b), the space is characterized by the presence of several distinctive valleys. Such a task is easy for solution, too. The most difficult situation occurs when there are a lot of nods and they are arranged in a relatively uniform manner (see Fig. 2c). The space of states is characterized then by a large number of vast and flat valleys, which differ insignificantly in their depths.

![Figure 2. Different ways of the topology of the nods](image)

We can notice a fact that in the case of a uniform arrangement there exist groups of solutions completely different from one another from the point of viewing the order of displacement, and which differ insignificantly in their total weights. Such systems are difficult to solve both with intuitive methods and with known algorithms.

The function of both neural networks and evolutionary algorithms is based on a search, random to certain degree, of the spaces of states and slow fixing in a certain local minimum. These algorithms in their initial phase of functioning through aggressive learning forced by a large learning coefficient or a great number of various individuals (in the case of genetic algorithms) while exploring the space of states orientate themselves on one of its valleys. This orientation is random to a large degree and is deprived of intelligence. Only after learning coefficients have been decreased algorithms orientate themselves on finding one of local optimums, however searching a different (and perhaps better) part of the space of states is no longer possible. Algorithms of a local search type are very effective in improving initial results, however very often they are time consuming. One can observe a varied effectiveness of particular algorithms in the cases of different types of nod arrangements. For this reason, a topological analysis of the
task in question constitutes an important element in the proper selection of an algorithm and its coefficients. Considering the abovementioned facts, the authors have elaborated a hybrid system based on a number of algorithms to solve the tasks in question.

ARRANGEMENT OF NODS: ITS INFLUENCE UPON SELECTION OF ALGORITHM

From the practical point of view, the system in order to be flexible and universal should adapt itself to the set of input data. Additionally, the system should be provided with various criteria of alloy and a library of algorithms used by the solver module being as large as possible. The effect of the work of the system should be a solution fitting the imposed limitations of time and the solution quality. As shown in Fig. 3, the system designed by the authors comprises the following blocks:
- the module of input data analysis – the purpose of this module is the analysis of input data, which comprises a statistical analysis, computation of the upper value and the lower limit of the set of solutions,
- base of solutions – a set of problems which have already been processed, as well as of the best solutions found,
- base of typical systems – is a set of typical and regular systems, for which we know optimal solutions,
- module of selection of algorithms and parameters – this is a module making use of artificial neuron networks and an expert system, which on the basis of the data obtained from the of input data analysis module as well as time limits and quality, selects proper algorithms and their parameters in order to construct a hybrid system,
- the constructor of a hybrid algorithm – this is a module which on the basis of data received from the module of the selection of parameters constructs the proper sequence of algorithms and of their parameters,
- a library of algorithms – this is a library of algorithms available in the system,
- the solver – this is a module in which an algorithm constructed processes data in order to obtain a solution. The solver generates input data in the form of a solution weight and the time of the computational unit necessary for this. In the training phases these data return to the module of the selection of parameters in the form of a feedback for the purpose of neuron learning. On the input, the system receives raw input data in the form of the coordinates of nods and possibly the required solution quality, as well as the maximum time assigned for processing.

![Diagram](image)

Figure 3. Scheme of the system

Next, these data go through the statistical analysis module, in which a normalization, a statistical analysis and an analysis of concentrations are conducted. Then the database of solutions and the database of typical systems are searched in order to possibly find satisfactory solutions. If such a solution does not exist the statistical analysis module passes the generated information on to the module of selection of algorithms and parameters. The next stage comprises the construction of a hybrid algorithm. This is done in the algorithm constructor, which on the basis of parameters takes out the methods from the library of algorithms. Once a hybrid algorithm has been constructed it finds its way to the solver, where the final processing occurs in order to find the solution.

STATEGIES OF ALGORITHM SELECTION

In the library of solution methods available in the system there are such algorithms as the following: Kohonen’s artificial neuron network, a flexible network, network, an evolutionary algorithm, as well as a few new and own methods of local search (Szatkiewicz 2002, Królikowski 2001). All the algorithms have been provided with original parametering procedures, which increases their flexibility. In networks making use of Kohonen’s algorithm and the flexible network, a dynamic selection of the number of neurons and an automatic selection of learning coefficients have been applied.
As experimental examinations have shown, very good effects are achieved in many cases through a division of a task into sub-tasks while applying an analysis of concentrations. Then every separated cluster is solved individually and in the final stage these are joined in a global solution. This has a particular significance for tasks with a large number of nodes, in which a reduction to sub-tasks with smaller sizes significantly shortens the time of computations. Additionally, it has been observed that algorithms based on artificial neuron networks are relatively fast, but solutions as generated by them are characterized with a large scatter and a poor quality.

In order to finally improve the solution a decision was made to apply local search algorithms, which have a bit greater computational complexity but also they very effectively search a selected fragment of the space of states finding the optimum in a defined neighbourhood. Individual stages of the solution search are shown in Fig. 4.

The system while automatically analysing the solution state selects the proper and the most efficient algorithm on a given stage of search. During the learning process of the system it was observed that Kohonen’s artificial neuron network is chosen most frequently as an initial optimiser and, in the final phase, sequential procedures of local search.

SYSTEM EFFICIENCY

The basic assumption for the system was that it be flexible and efficient. The system’s efficiency consists in the fact that it should solve tasks of an optimisation of a displacement trajectory in a wide range of extents and topologies of the problem. Efficiency requires from the system fast and good quality solutions. An important parameter of the system’s work is a reasonable selection of the values of time limitations and the weight of the solution. These are of course contradictory criteria and an unreasonable selection of these may result in a situation when there exists no possibility to solve a given problem with the assigned criteria.

The system was implemented in the form of an application written in C++ language in the Windows environment and tested on various sample problems with the extents ranging from 100 to 1000 nodes. Depending on the extent of the task and the required solution quality, the system finds solutions in time shorter than 1 sec. on a processor of P4 class and the clock of 1.4 GHz for problems with 1000 nodes. If the system has a lot of time for finding the solution it tends to repeat several times the work of the neuron network and a modification of solution through the usage of more time-consuming local search algorithms with large parameters of the search depth.

The system has met the design assumptions both in cases when we need very fast results and also in situations when we have more time for computations and expect very good quality solutions.

REFERENCES


Osowski S. 1996 “Sieci neuronowe w ujęciu algorytmicznym” WNT Warszawa

Sysło M.M. 1993 „Algorytmy optymalizacji dyskretnych”, PWN Warszawa

Szatkiewicz T. 2002 „Hybrid system for minimalization of trajectory of displacement based on neural networks and tabu search procedures” Polioptymalizacja i komputerowe wspomaganie projektowania WNT Warszawa

Tadeusiewicz R. 1993 „Sieci neuronowe” Akademicka Oficyna Wydawnicza, Warszawa

BIOGRAPHY

TOMASZ SZATKIEWICZ was born in Koszalin, Poland and went to the Technical University of Koszalin, where he studied fine mechanics and information theory and obtained his degrees in 2000. He is a Ph.D. student at Department of Fine Mechanics at Technical University Of Koszalin. He is interested in artificial neural networks and optimisation theory.
A MULTI-ECHELON SYSTEM’S SIMULATION MODEL FOR REPAIRABLE AND CONSUMABLE ITEMS MANAGEMENT

M. Elisa Cunha
Dep. Ciências e Tecnologias
Univ. Autónoma de Lisboa
R. Santa Marta, n°. 56
1169-023 Lisboa, Portugal
E-mail: melisa@ual.pt

Ana Paula Póvoa
Dep. Economia e Gestão
Instituto Superior Técnico
Av. Rovisco Pais
1049-001 Lisboa, Portugal
E-mail: apovoa@ist.utl.pt

J. Assis Lopes
Dep. Economia e Gestão
Univ. Lusíada de Lisboa
R. da Junqueira, nº. 188
1349-001 Lisboa, Portugal
E-mail: alopes@lis.lusiaida.pt

KEYWORDS
Inventory management, repairable items, multi-echelon systems, simulation, logistics, stochastic models, decision aid systems

ABSTRACT
Repairable items inventory management is complex and important. On the one hand, complexity is due to a variety of reasons such as system structure, unknown repair times and multi-indenture item configuration. On the other hand, importance is traced since stock determines systems operability.

Several multi-echelon inventory models for repairable items have been developed, however even the most recent have restrictions to their successful application.

To surpass some of these limitations a simulation model is developed in this work. This not only permits the explicit consideration of hierarchical parts structure but also admits a multi-echelon structure system (three levels). Depending on the failure, reparation can be done at different places and the repair time depends on the components’ stock or, if there are none, on the repair time of the missing items.

The model establishes a relationship between the system’s operability and the stock of the different parts. An $(s,Q)$ or $(s,1,S)$ inventory policy is assumed for consumable or repairable items respectively.

The resulting model is quite general where no restrictive assumptions on the cases to model are required.

INTRODUCTION
Stocks are an important investment in all economic activity areas. The investment is greater if the material to be managed is repairable (items with a higher per-unit price, a wider life cycle and a lower repair cost than the acquisition cost of a new item). This fact points to the importance of repairable items comparatively to consumable ones. However, this relevance depends, also, on system involved.

Repairable items management is very complex due to a variety of reasons. Firstly, the items usually belong to multi-echelon structure systems. On the lower level are the bases, which are supplied by one or more depots fixed at an upper level. Note that the word supply can be used in three different meaning: an immediately upper echelon demand for a reusable item, an external supplier new item’s purchase or a failure item being sent to repair facilities. Secondly, the unknown repairable time and supply time implies the usage of probability distribution to characterise these parameters. Usually, it is indispensable to define different probability distributions: one to identify the repairable time, depending on the failures and resources at the repair facility, other to establish the interval between the order’s base to the depot and the instant of time where the item is available at the base, if the depot has components on stock and a third distribution is needed to characterize the elapse time for when an item is available at the depot. Finally, items’ multi-indenture structure implies a simultaneous optimisation for all item’s components.

Several multi-echelon inventory models for repairable items have been developed and published (Guide and Srivastava 1997). However even the most recent models have restrictions to their application. For example, the majority of models’ complexity is responsible for its non-application in real case studies therefore the METRIC (Sherbrooke 1968) model utilization is going on, although it underestimates items’ backorders (Diáz 2003). On the other hand, none of the methods simultaneously allow a small number of items exposed to failures and limited repair facility (Rustenburg et al. 2001). Another gap is finding when it is attempted to manage repairable and consumable items at the same time.

To surpass some of these restrictions we have developed a simulation model to manage repairable and consumable items in a multi-echelon system where a repairable items’ structure is considered explicitly. A simulation technique was used which allows the development of a detailed model with no need of simplifying and disarranging restrictions inherent to analytical models. Also it enables different probability distributions to be used at important decision variables.

SYSTEM TO MODEL
The system considered has a three-echelon structure: one base in the lower level, one depot in the upper level and one intermediate echelon in the middle level (Figure 1). Each level has a repair and a storage facility. The later stores all components requested by the reparation done in the echelon or required so as to satisfy the immediate lower echelon demand. The repair facility has a limited capacity imposed by the number of technicians and repair lines. A reduce number of working items are associated
with the base and the items’ hierarchical parts structure can be expressed explicitly. For example, a final item is formed by a set of major assemblies (first level) and these may consist of several components (lower levels components), subordinated to the assembly and parts hierarchy. Components can be accessed without dismounting the item, if that describes the reality.

![Diagram](image)

**Figure 1: System to Model Structure**

An item is subject to failures caused by consumable or repairable components mal-function. Depending on the failure, repair can be done at the base or at the intermediate echelon. When arriving at the base, an item waits in a FIFO queue until its repair can be started. Item failures are checked and a decision on the repair local is taken (the base or the intermediate echelon). If at the base the malfunctions components are dismounted and dispatched to the repair facility. A request is made for a repairable or a new component to the echelon storage facility. If at the intermediate echelon, items repair takes a long repair time since it implies items disassembly. Therefore, all first level item assemblies are replaced by repairable items independently of being or not damage.

Items repair time depends on the components or assemblies’ stock on the shelf or, if no stock is available, the elapsed time for a component or assembly to be at the echelon. In this case, it can arrive from the echelon repair components’ facility or from the immediate upper echelon storage facility.

Depending on the failure, components can be repaired at the base, at the intermediate echelon or at the depot. Like items, the components wait at the repair centre in a FIFO queue. Technicians check components’ failures and decide if the repair can or cannot proceed in the place where the component is. If technicians decide to send it to the immediate upper echelon a mended component is delivered from the upper echelon storage facility. Components repair time depends, also, on the lower components’ stock at the echelon where the repair is done or, if there are none, the elapsed time for a component is available. In this case, the component can arrive from the echelon repair components’ facility or from the immediate upper echelon storage facility.

When the reparation is finished the item is sent to work and the component is sent to the storage facility of the echelon where it has been repaired or, if there are backorders, one is satisfied.

Whenever the damage does not justify the repair, items or components must be discarded. This situation imposes an obligation to define, by the depot, a policy for items and components acquisition, based on a hierarchical information system.

The model’s objective consists of establishing a relationship between the maximum stock inventory position, for every components, and the system’s operability, estimated from all items that have non-work conditions.

**ASSUMPTIONS**

The following conditions are assumed for the model:

- Failures are independent and identically distributed;
- Demand is satisfied on a “first came first served” basis;
- Final items, assemblies and components failure transportation times are independent and identically distributed;
- Items and components repair times are independent and identically distributed;
- Transportation times for new or repairable components needed in reparation is included in repair time;
- Transportation or reparation times can follow an empirical or a theoretical distribution such as: Beta, Binomial, Erlang, Exponential, Gamma, Geometric HyperGeometric, Uniform, Normal, LogNormal, Poisson, Triangular, Weibull distribution;
- A (S–n, S) policy, where n represents the number of components dismounted, is used either on demand of mended components or on the components sent to repair;
- Acquisition policies are (s, Q), for consumable components, and (S–1, S), for repairable one.

**THE MODEL**

The model was built using the Extend™ (1998) package that has inherent an object-oriented simulation, with predefined object frames that describe various simulation requirements. Those objects are called blocks and are composed of an icon, connectors and a programme. The icon is the graphical representation. The connectors are represented as small squares localized around the icon and through which blocks are linked and the information flows. There are two types of information involved; one concerning the moving objects along the simulation model.
that can have associated attributes and priorities, the other about values that can change during the simulation run. The programme defines how information is used and transformed by the block.

The modeller may use the pre-defined objects specifying their behaviour or may create new objects by using the ModL, Extend™ object-oriented language based on C++.

The model developed is a discrete simulation model and the simulation time increment is based on the following events: a working item’s failure; an damaged item beginning or ending reparation at the base or at the intermediate echelon; a damaged component starting or concluding reparation at any echelon; an item or component arriving at the repair centre.

The model considers a system with a hierarchical structure as shown in Figure 2. Six different types of blocks are considered and these are respectively the “Executive” block, the “External Supplier” block, the “Working Items” block, the “Base” block, the “Intermediate” block and the “Depot” block. Items’ movements are allowed from and to the different blocks and are represented in Figure 2 by oriented segments. The different blocks characterization is the following.

**“Executive” block**

The “Executive” block is a compulsive block in all Extend™ discrete simulation models, being responsible for time increment.

**“External Supplier” block**

The “External Supplier” block represents the system external suppliers to whom the depot acquires news items or components. As it is an external system entity, its tasks are not considered and its presences in Figure 2 is justified since it represents the link between the system and its external environment.

**“Working Items” block**

The “Working Items” block symbolizes items in their activity at the working place. It is a hierarchical block so it is composed of other blocks, those depicted in Figure 3.

![Figure 3: Working items block structure](image)

Analysing the Figure 3 we can see on the top left the “Base_In” and the “Intermedia_In” blocks through which items arrive from the base and the intermediate echelon repair facility, respectively.

Considering for illustrative purposes a single working item is assumed and represented by “Working Item” block. The item can breakdown due to wearing.

Based on failures’ item probability distribution function defined in the “DamagedDist” block, it is asked if the working item has broken, the “Damaged Item” block. If the test result is negative, block connector identified by the character “N”, the item goes to operation through the “Throw” block. Otherwise, block connector identified by the character “Y”, the simulation proceeds to the identification of damage component, the “Which comp. is failed?” block. This analyses is based on the conditional probability distribution function of an item’s failure caused by one component. Depending on the failed component, the item’s repair place is chosen, the “BaseRepairPlac” block, which may be the base, the “Base_Out” block, or the intermediate echelon, the “Intermedia_Out” block. The transport is symbolized through the “Base Transp” or the “Intermed Transp” blocks, respectively. In either case, transportation time is generated based on the transportation time distribution function, “TranspTime” blocks.

**“Base” block**

The “Base” block in Figure 2 can be decomposed in a set of hierarchical blocks (Figure 4) representing the base tasks.

Through the “WorkPlace_In” block the damaged items arrive and wait in a FIFO queue represented by the
“Damaged Items” block. When necessary conditions are verified the item enters in the repair line through “Gate” block. Firstly, failures are checked in the “FailureConfirm” block which processing time is generated by a probability distribution, the “Confirm Time” block. Then and as a function of the repair’s type the “Repair Place?” block sends the items to the appropriate place. If repair is impossible at the base, item is sent to the intermediate echelon from the “Item Out” and the “Intermediate Out” blocks. If possible, repair proceeds at base, the “Item repairation” block, which represents all tasks related with the item repair. These tasks are not represented so as not to overload the Figure but can be resumed the following way. At the beginning, the damaged item components are dismounted and sent to the components repair place, “Component reparation” block. Simultaneously an order, for all components needed, is posed to the echelon storage facility. If there is stock the components are sent immediately; if not the demand is backordered until there is an available component which may arrive from the base component repair facility or from the intermediate echelon storage facility. When the repair is finished the item is sent to the working place through the “WorkPlace_Out” block and a message indicating that a line is free (the “Free 1” and the “Free 2” blocks) is sent to the “Gate” block.

![Base block structure](image)

The “Component reparation” block performs similar to the “Item reparation” block then when the components arrive at the base repair facility they wait in a FIFO queue, until a repair component line is free. Failures are checked and the repair place is decided, the base or the intermediate echelon, depending on the failure’s gravity. Check time is generated from a probability distribution. If the failure cannot be repaired at the base the component is sent to the intermediate echelon components’ repair facility, the “Comp_Out” and “Intermediate_Out” blocks; transportation time is generated from a probability distribution function. If the component can be repaired at the base, spare parts are ordered to the base storage facility. If all spares are in stock the storage facility sends them immediately, otherwise the demand is backordered until a spare arrive from the base components repair facility or from the intermediate echelon storage facility. When the repair is finished all components are sent to the echelon storage facility or if there are backorders one is satisfied.

Note that although a single line has been described several ones can be simulated in the model using the same methodology of blocks structure.

“Intermediate” block

The “Intermediate” block from Figure 2 models the intermediate echelon and the associated tasks as represented in Figure 5.

![Intermediate echelon block structure](image)

Figure 5: Intermediate echelon block structure

An item arrives at the intermediate echelon from the working place or from the base by the “Item_Work_In” or the “Base_In” blocks, respectively. It waits in a FIFO queue, the “Damaged Items” block, until a repair line becomes free. When the necessary conditions are verified the item enters in the repair line through the “Gate” block. Firstly, the first level item components are dismounted, the “Item Dismount” block. Secondly, two simultaneous tasks occur, the “SimultTasks” block: (1) all first level item components are ordered to the echelon storage facility, the “Demand” block; (2) components retired are sent to the components repair facility, the “Component repairation” block. If the echelon storage facility, the “Storage facility” block, has stock demand is immediately satisfied; if not the demand is backordered until there is an available component arriving from the echelon component repair facility or from the depot storage facility. When arriving, the “Receipt Comp” block, the required components are mounted in the item, “Item Mount” block. After repair, the item is sent to the work place through the “WorkPlace_Out” block and a message is sent to the “Gate” block indicating the free stage of the line.

The “Component reparation” block represents all tasks related with the components repair. As before, they are not symbolized in the Figure but they can be resumed as following. When the component arrives at the intermediate component repair facility waits in a FIFO queue, until a repair component line become free. At that time, the failures gravity is checked and the repair place decided, the intermediate echelon or the depot. Check time is generated from a probability distribution. If the failure cannot be repaired at the intermediate echelon the component is sent to the depot components repair facility; transportation time is generated from a probability distribution function. If the component stays at intermediate echelon, all spare parts needed are ordered to the echelon storage facility. If all spares are on stock the storage facility sends them immediately, otherwise the demand is backordered until there is an available spare sent from the components repair facility at the intermediate echelon or from the depot. When the repair is finished all components are sent to the echelon storage facility or if there are backorders one is satisfied.
“Depot” block

The “Depot” block in Figure 2 represents the depot and it is a hierarchical block composed by the blocks shown in Figure 6.

![Figure 6: Depot block structure](image)

Looking at Figura 6, when a component arrives at the depot repair facility, through the “Intermediate_In” block, it waits in a FIFO queue, the “Damaged Comp” block, until there are technicians to check the failure extension and to decide if the component can or cannot be repaired, the “Reparable?” block. Check time is generated from a probability distribution, the “CheckDistrib” block.

If the failure can be repaired the component waits again into another FIFO queue, the “ReparableComp” block, for a free repair line. When the necessary conditions are verified, the component enters the repair line through the “Gate” block and its reparation starts, the “Component reparation” block. The component elements are separated and sent to the repair centre. After that, an order is posed to echelon storage facility on the spares to assemble the component. If there are spares on stock demand is satisfied, otherwise it is backordered until there is a spare available arriving from the repair centre or from an external supplier, if a replenishment demand had been done. When the repair is finished all components are sent to the storage facility, the “Storage Facility” block, or if there are backorders one is satisfied.

If the failure cannot be repaired the component is sent to the scrap, the “Scrap” block, and it must be verified if a new replenishment must be done, the “New acquisition test” block. This verification is based in a hierarchical information system where the component type is the key to provide all indispensable information. Firstly, one unit is subtracted from the component stock inventory position. Secondly, if the component is reparable an unit size replenishment is ordered to the external supplier; otherwise, a Q units size replenishment is ordered if the inventory position has dropt to the reorder point or below that.

CONCLUSION

A generic model for the management of a multi-echelon system with repairable and consumable items was presented. This was built in an object-oriented simulation language, the Extend™ (1998), and establishes a relationship between the maximum stock inventory position, assumed for every component, and the systems’ operability estimated from all final items not operable. The model is able to represent real system with different levels of complexity where no restrictions hypotheses are required. It allows the establishment of different components relationships independently of the hierarchical level occupied in the system decomposition tree. Decision variables can be generated from different probability distributions and visualization of the modelled system performance is possible through the animation existent in the tool used (Extend™), which acts as a comprehensive model capability to the user / developer. Finally the model application was not explored in the present paper, due to the lack of space, but it will be published elsewhere, where a real case study of Portuguese Railway 1900 Locomotives will be solved.

REFERENCES


AUTHOR BIOGRAPHY

M. ELISA CUNHA is a Ph.D. student in the Dep. Economia e Gestão of Instituto Superior Técnico in Lisbon, this institution had given her before a M.Sc. degree in Operational Research and Systems Reengineering. She teaches Operational Research at the Dep. Ciências e Tecnologias of Universidade Autónoma in Lisbon.
AIRPORT
TRANSPORT
FLOW
SIMULATION
PASSENGER AND BAGGAGE FLOW IN AN AIRPORT TERMINAL: 
A FLEXIBLE SIMULATION MODEL

Lorenzo Brunetta 
University of Padova 
Via Gradenigo 6/A 
35131 Padova PD 
Italy 
E-mail: brunetta@dei.unipd.it 

Giorgio Romanin-Jacur 
University of Padova 
Stradella San Nicola 3 
36100 Vicenza VI 
Italy 
E-mail: romjac@dei.unipd.it

KEYWORDS
Airport terminal, Discrete simulation, Flexible model, User friendly interface.

ABSTRACT
The lack of a friendly and flexible operational model of landside operations motivated the creation of a new simulation model adaptable to various airport configurations for estimating the time behavior of passenger and baggage flows, the elements’ capacities and the delays in a generic airport terminal. The validation of the model has been conducted by comparison with the results of previous research about the average behavior of the future Athens airport. In the mean time the proposed model provided interesting dynamical results about both passenger and baggage movements in the system.

INTRODUCTION
The modeling of airport terminal operations has advanced significantly over the last 15 years; see for instance (Tosic, 1992) and (De Neufville and Odoni, 2003). Available models have improved in detail and reliability, as well as “user friendliness”. As a result, their use as decision support aids in management or design tools in terminal development projects has been steadily increasing. Some existing models are “strategic” in nature, as they sacrifice level of detail in exchange for speed and flexibility, and are employed for long-medium term decisions, while others are primarily “tactical”, incorporating high level of detail in data and system definition, and are employed for medium-short term decisions. Recent tactical simulation models evidence two main defects. Some of them, like for instance (Gatersleben and Van Der Wej, 1999) and (Joustra and Van Dijk, 2001) either model one airport but cannot be extended to other ones, or model one part of the airport. Other models, like for instance (Brunetta and Romanin-Jacur, 1999) get wide flexibility, i.e., ability of describing different airports in detail by means of limited adjustments, but lack user friendliness. The lack of a satisfactory tactical simulation model of landside operations motivated the creation of our new flexible simulation model for estimating the time behavior of passenger and baggage flows, the capacity and the delays in a generic airport terminal. Our model overcomes, on a low-cost platform, typical problems of the old packages, such as large data requirements and lack of flexibility. In fact it includes the common features of all airport terminals and, after an appropriate definition of the key parameters, is capable of describing any terminal configuration. All input data concerning airport structures and related operation times are inputted by means of guided menus, while flight plans are inputted by a text file. Any modification or adaptation is very easy, also for non expert users.

ENVIRONMENT
In recent years the amount of passengers simultaneously present in airports has dramatically increased because of many different causes. On one side the risk of attempts does not reduce significantly the amount of travellers by plane, but at the same time it imposes strong actions against terrorism taking place in the airports or originating from them; such actions require strict checks on passengers, baggage and goods to be boarded, which lower all movements inside the terminal, and consequently oblige passengers to extend their stay before boarding. On the other side significant changes in air travel management were imposed by the necessity of strong savings for airline companies. Many companies adopted the hub and spoke system, which replaces point-to point direct flights with a series of two or more indirect ones, linked together at the company hub airport (or airports); such a system plans several flights converging to the hub airport, linked to other flights originating from it, with a short time between arriving and starting ones; as a consequence large flows of transit passengers affect the hub terminal in some well defined intervals of the day, especially in the early morning and in the late afternoon; moreover the system permits to correctly dimension employed aircraft and therefore to obtain a high utilisation index. Code sharing between airlines, as well as charter and low cost companies, succeed in filling aircrafts, and often encourage the use of larger aircraft. All abovementioned changes contribute in increasing the amount of passengers simultaneously asking for the same facilities (ticket counters, check-in desks, security checks, baggage conveyors, baggage sorting stations, baggage inspection station, etc.), especially during peak intervals of the day. It is obvious that in this hard situation all (technological and human) resources are particularly stressed. In order to contain costs and yield efficiency (avoid congestion, i.e., delays, queues and bottlenecks) a clever utilisation of structures and facilities (and related servers) is required, possibly sharing some of them among different users.
MODELING APPROACHES

A correct management of an airport terminal may be considered from two different points of view. A first approach concerns resource dimensioning based on the mean behaviour of the system: here we consider the “typical” days we foresee in the year (in particular the “busiest day”), by taking into account the related peak intervals. Resource activation and flow ruling are “statically” decided, in order to obtain acceptable levels of service, based on the above estimates. Obviously we adopt a deterministic and aggregate model, which is generally utilised to determine the total amount of resources at disposition; the main advantage of such a model is its quick response and its wide flexibility, as it may be easily adapted to different realities; on the contrary, as it solves static dimensioning problems during predetermined intervals, it is able neither to describe transient evolutions of the system between adjacent (quasi) equilibrium states, nor to reproduce the system behaviour consequent to any alteration of the inputs (see for instance delays and similar).

A second alternative (and sometime complementary) approach is addressed towards a detailed description of the system behaviour, consequent both to normal and to exceptional situations. As an example, we may want to examine the system evolution in time, arising from a temporary interruption of air services due to fog, from a congestion due to delay of one or more flights, etc., and thus to plan in advance, by accurate scheduling, all suitable reactions to reduce disease and to recover effectiveness and efficiency. As a second example, remember that the whole system simultaneously manages passengers, baggage and goods, to be loaded, unloaded or transferred from an aircraft to another; they generally follow different routes and utilise different facilities, but they meet in many common points and interact; possible delay or unsatisfactory behaviour of one of the subsystems may affect the whole system behaviour, causing further delays and deadlocks: we may want to foresee, monitor and rule all subsystem interactions to avoid delays and/or inefficiencies, or anyway to reduce system sensitivity to random accidents which may trouble it. In order to satisfy all abovementioned requests the only answer is supplied by discrete simulation.

THE SUGGESTED MODEL

A useful airport terminal simulation model should describe faithfully the particular airport under study; a model specifically built up and implemented only for that surely satisfies such a request, but it presents the limit of being hardly adaptable both to possible future transformations and expansion, and to other airports, due to all peculiar characteristics; in other words, it is a heavy job to obtain an instrument to be employed only a few times, to satisfy particular exigences. On the contrary, a generalised flexible model, describing the common structure of all airports, and built up to be easily parametrised in order to model different airports, requires about the same initial effort but permits to be employed many times, both for different airports and for different operating conditions. The aim of the paper is a flexible model implemented by a user friendly package. The model describes in detail all operations effected by both passengers and baggage; in particular we consider: passenger generation; departing, arriving and transit passenger movements in the terminal, with distinction between Schengen/domestic and extra Schengen/international passengers; departing, arriving and transit baggage movements by means of the baggage handling system.

Flight departures and arrivals take place following the planned schedule (obviously delays may be considered); according to the flight type (arriving or departing flight, border crossing or non crossing, aircraft capacity and saturation, passenger characteristics), every flight generates passengers according to a known passenger presentation time distribution, generally a beta type one. Departing passengers may need to buy tickets, choose a check-in desk (according to the number of open desks and to given rules), ask for check-in and simultaneously may generate baggage, cross security control blocks, move towards the waiting lounge, are assigned a gate, move to the gate and are boarded; if extra Schengen/international they follow a different route after checking-in and have to cross passport control blocks too; late passenger may follow a special route, if possible, related to the flight departure time. Arriving passengers on their own enter the arrival gate and reach the baggage claim area; if extra Schengen/international they have to follow a different route crossing passport control and customs. Transit passengers constitute a fraction of arriving passengers: after entering the arrival gate they move towards the right waiting lounge to join departing ones. Obviously passengers may form queues whenever they ask for a scarce resource, like desks, security check blocks, passport control desks, etc. Departing baggage, coming either from the check-in desks, cross x-ray machines (all baggage or a fraction, according to the airport rules) and possibly further inspection apparels and/or operators if necessary or convenient, then they are sorted and sent to the right pier to be loaded; particular (irregular, oversized, etc.) items are treated apart. Arriving baggage are unloaded, moved to the inbound area, sorted and then sent either to the baggage claim area (arriving baggage), or to the transit baggage area (transit baggage), to be possibly relabeled and finally moved to join departing baggage. Baggage may form queues whenever a scarce resource is to be employed.

MODEL IMPLEMENTATION

The proposed model is implemented by package MicroSaint 3.2 on a common use personal computer; MicroSaint showed to be a good compromise between abstraction and user friendliness. In particular it revealed to be easily readable also by non expert users.

The model implementation include 51 tasks, as may be seen in Figures 1-4, where the main model network and three subnetworks are reported; it is comparatively simple and synthetic, but however it represents an actual airport in detail. There every task corresponds to a well defined operation, as evidenced by the related box name, and contains all information about: 1) possible queue for entering entities
(aircraft, passengers or entities) and related rules (e.g., priority); 2) beginning conditions (e.g., required resource at disposition) and beginning effects (e.g., resource sizing); 3) time to perform the operation (probability distribution); 4) ending effects (e.g., resource releasing); 5) decision about outgoing entities, which may be either multiple (i.e., the entity is multiplied and every new generated entity follows a different route) or probabilistic (a single route is chosen with given probability) or tactical (a single route is chosen according to a given deterministic rule); obviously information include the related parameters. Therefore all airport functional characteristics are stored in the related tasks, together with possible time and/or state depending operating rules. All input data concerning arriving and departing flights (type of aircraft, number of passenger and related baggages, aircraft arrival or departure time, distribution of the arrival time to the airport terminal for departing passenger, etc) are supplied by a text file, to be written in the “Input data” 1.2 box in Figure 2. Such a way all data which are necessary to define both the airport characteristics and the planned operations are easily and clearly set and may be written in plain language.

Model outputs include: delays on departure, flight by flight, distinguished between those caused by passengers and those caused by baggage; last passenger check-in time, flight by flight (departing passengers); last passenger leaving the airport time, flight by flight (arriving passengers); queue length distribution, maximum queue length, queuing time distribution maximum queueing time, at every facility. Obviously simulation may be effected with different amount of open facilities and/or with different operating conditions and rules, both normal and exceptional (e.g., consequent to any alteration of flight schedule, due to bad weather conditions, strikes, etc).

APPLICATION RESULTS

The suggested simulation model was employed to describe Athens International Airport, newly built for the Olympic Games, currently not yet completed.

The airport model was tested under normal conditions, corresponding to 508 aircraft movements per day under the planned operating conditions.

All average simulation results were positively checked by comparing them with the ones, reported in (Andreata et al. 2001), supplied by aggregate deterministic model SLAM, described in (Brunetta et al. 1999).

The results revealed that the airport is well dimensioned and possible departure delays or decrease from high values of service levels recommended by International Air Transport Association (IATA) are rare and last for very short time intervals. However it was possible to suggest some small changes in the number of open check-in desks during well determined intervals of the day, by which a strong reduction in the departure delays and in the passenger queueing times can be obtained.

The model will be employed soon for other applications.

REFERENCES


The case of Athens International Airport”. In Proceedings of Odyssey 2003. CD, Palermo, Italy.
De Neufville R. and A. Odoni. 2003. Airport system planning, design and management, Mac Graw Hill, Columbus, OH, USA.

BIOGRAPHY

LORENZO BRUNETTA got a “Laurea” degree (equivalent to Master of Science) in Mathematics in 1991 and a PHD in “Computational Mathematics and Computer Science” in 1995 both from the University of Padova. From 1996 to 2001 he was Assistant Professor at the Engineering School of the Polytechnic of Milan. He is currently Associate Professor of Operations Research at the University of Padova. He collaborated at several EU, NATO, and MIUR Research Projects. He is coauthor of over 40 publications including fourteen in International journals. His main research interests are Combinatorial Optimization and Integer Linear Programming and their applications to Air Traffic and Telecommunication Systems.

GIORGIO ROMANIN-JACUR got a “Laurea” degree (equivalent to Master of Science) in Electrical Engineering in 1970 from the University of Padova. 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. He collaborated at several EU, CNR, and MIUR Research Projects. He is coauthor of over 100 publications including 28 in International journals. His current researches concern production scheduling in the fields of Industry and Services.

363
Figure 2: Passenger generator subnetwork

Figure 3: Baggage Handling System subnetwork for departing baggage

Figure 4: Baggage Handling System subnetwork for arriving baggage
NEW CONCEPTS IN BAGGAGE OPERATIONS
FOR NARROW BODY AIRPLANES
A Simulation Study

J. A. Ottjes and J.C. Rijsenbrij
Delft University of Technology, Faculty of
Mechanical Engineering and Marine Technology
Mekelweg 2, 2628 CD Delft, The Netherlands
J.A.Ottjes@wbmt.tudelft.nl

E.B.C. Koelemeijer and B.M. Snick
Amsterdam Airport Schiphol
Business Unit Airlines
Schiphol Group
Evert van Beekstraat 202, 1118 CP Schiphol

KEYWORDS
Airport, baggage, discrete simulation

ABSTRACT

In this paper new concepts for baggage transport to and from narrow body airplanes are presented. The concepts are simulated and applied to a part of Schiphol Airport Amsterdam. The current baggage transportation is labour intensive and bares the risk of damaging or losing bags. Moreover it is time-critical because of the tight flight schedules used. An alternative scheduling method as well as the application of a partly automated baggage loading and unloading vehicle are investigated with simulation and reported. It appears that a considerable saving is possible when using both the scheduling method and the new baggage vehicle.

INTRODUCTION

Amsterdam Airport Schiphol is a so-called “one terminal” airport. It consists of one building for the passengers with short connections to the gates. Figure 1 shows the current and planned future situation. The B-, C- and part of the D-pier form a separated “Schengen area” for destinations in the European Union. Schiphol handles about 40 million passenger per year 42% being transfer passenger just changing plane at Schiphol. It is anticipated that air traffic via Schiphol will grow 4 to 5 % per year in the next decennium. This will have serious consequences for baggage handling such as congestion on the infra structure and heavy load on the baggage sorting systems. The existing piers are rather close to the baggage sorting system but distances to new piers such as the J-pier will become considerably larger. Moreover changing regulations concerning irregular labour schemes and heavy physical working conditions are expected to cause difficulties in manning the baggage handling service.

BAGGAGE OPERATIONS

Baggage for wide body aircrafts is contained in containers. Narrow body planes, for example the Boeing 737, are too small for standard air containers (ULD’s) and are loaded by hand, bag by bag into/out of the belly, see figure 2. This paper is restricted to baggage flows to and from narrow body planes.

Figure 2: Unloading of a Narrow Body Airplane onto a Baggage Cart.

Current Procedures

Currently baggage for narrow body airplanes is transported using trains consisting of up to six baggage carts pulled by a manned truck. This system is flexible and, provided there are enough baggage cars available, the number of pull-trucks can be minimized. Some disadvantages of the systems are a low velocity (loaded up to 15 km/h), a certain risk of losing or damaging baggage and labour-intensive and physically heavy work with respect to loading and unloading. (Jerkovic 2000)
Arrival
If an airplane arrives, a central coordinator assigns a vehicle (train) and a handling team to the gate of the arriving plane. The team unloads the arriving baggage assisted by the truck driver. The train is then driven to the transfer and the reclaim inlet. The loading team returns to its base to get further instructions. Only if the turnaround time of the plane is short enough, the loading team remains at the gate to load the plane before departure.

Departure
At a certain time before scheduled departure the central coordinator sends a baggage train to the baggage sorter outlet assigned to the departing flight. There the baggage is loaded onto the baggage train and transported to the proper gate. The coordinator is also responsible for directing a loading team to the gate. The loading team, assisted by the driver, loads the baggage into the plane piece by piece.

CONCEPTS
In this paper three aspects will be studied, (Snick 2002)
– Introduction of a baggage truck (Oosterhuis 2002)
– Improving scheduling of personnel and equipment by dynamic planning.
– Influence of distance between baggage outlet and gate.

The Baggage Truck
The design of the baggage truck is shown in figure 3. Baggage transport per baggage truck offers the next features:
– The baggage truck can be loaded/unloaded automatically at the sorter outlet and reclaim and transfer inlet respectively.
– It lifts its baggage outlet to the level of the plane-belly entrance. There the baggage has to be stowed into the plane-belly by hand. Arriving baggage has to be retrieved manually from the belly, but from that point the further procedure is automated again.
– The load and unload rates are rather high and all baggage of one narrow body plane fits in the baggage truck.
– Connection with existing landside systems is possible.
– First In First Out (FIFO) loading and unloading per belt conveyor is possible.
– Odd-size baggage can be transport separated from regular baggage with the same truck.

Improving Scheduling
In the current practice there is no communication between central planning and labour teams and vehicles during operations. After each job completion personnel and drivers go to their base called the “buffer” and report to the central planner.

In the concept of dynamic planning (un)load teams as well as equipment drivers have radio contact with the planner and are assigned to jobs without physically moving to the

buffer. A procedure has been applied to reduce the total equipment travel time, maintaining the punctuality and service rate, see table 1. For each combination of job and available vehicle a time factor tij is calculated being the sum of the next three times: 1: time to next job, 2: penalty time if the vehicle comes from the buffer, 3: penalty time if the vehicle will arrive too late at its destination. The jobs

<table>
<thead>
<tr>
<th>Part List</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Belt Conveyor</td>
<td>Max. 175 bax</td>
</tr>
<tr>
<td>2. Lifting mechanism</td>
<td>20 bax/min loading</td>
</tr>
<tr>
<td>3. Shifting mechanism</td>
<td>36 bax/min unloading</td>
</tr>
<tr>
<td>4. Shifting mechanism</td>
<td>Max. 80 km/h</td>
</tr>
<tr>
<td>5. Elevating conveyor</td>
<td>12 m length</td>
</tr>
<tr>
<td>6. Conveyor drive</td>
<td>4 m high</td>
</tr>
<tr>
<td>7. Storage odd-size bag</td>
<td>2.6 m wide</td>
</tr>
<tr>
<td></td>
<td>0.2 m ground space</td>
</tr>
</tbody>
</table>

Figure 3. Baggage truck for transport and handling of baggage from and to airplanes. Patent Nr: 1022640 pending

are sorted in decreasing due-time so that the most urgent jobs are on top. The objective in the scheduling process is to reduce the sum of the time factors over the job-vehicle combinations and to maintain punctuality of service. A simple heuristic approach has been applied. For each job the vehicle with the smallest time factor in its corresponding row is assigned starting with the first job. In that way the most urgent jobs have the best choice of

<table>
<thead>
<tr>
<th>Vehicle Job</th>
<th>1</th>
<th>2</th>
<th>j</th>
<th>m</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>t1</td>
<td>t2</td>
<td>t3</td>
<td>t4</td>
</tr>
<tr>
<td>2</td>
<td>t5</td>
<td>t6</td>
<td>t7</td>
<td>t8</td>
</tr>
<tr>
<td>i</td>
<td>t1</td>
<td>t2</td>
<td>t3</td>
<td>t4</td>
</tr>
<tr>
<td>n</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Scheme of Time Factors for each Job-Vehicle Combination. Jobs are Sorted in Decreasing Due-Time.

vehicles. The schedule is refreshed after each job completion and plane arrival. The same approach is applied for the scheduling of the individual labour teams to the loading and unloading jobs of airplanes.

Influence of Distance
In the Schiphol situation the distances between terminal and gates are rather short as a result of the compact way of building and the "one terminal" concept. This situation changes if the airport is expanded with for example the new pier J at the other side of highway A4, see figure 1. In order to study the consequences a short distance situation (pier D) as well as a long distance situation (the future pier J) have been simulated. The distances between pier D and sorter outlet and baggage reclaim inlet are in the order of 100 m. For the J pier the distance to sorter outlet and baggage reclaim amounts up to 1500 m.

![Figure 4: Part of the Animation Screen showing three Planes being handled at Gates 6,7 and 8 at the D-pier](image)

**MODELING**

The process interaction method is applied. In that method relevant element classes are distinguished (Zeigler 2000; Vecke 2000; Ottjes 2002). An element class may be passive or active in the simulation. In the latter case the class owns a process, describing the activities of the element as a function of time. The main classes distinguished are:

- gate
- gate planner
- flight
- job
- vehicle (baggage train or baggage truck)
- vehicle scheduler
- loading team
- loading team scheduler

Each class owns its specific attributes. The flight class for example is characterized by the next attributes:

- Airline
- Flight number
- Planned gate
- Planned and real arrival time
- Planned, expected and real time of departure
- Number of arriving baggage units, split up into originating and transfer baggage
- Number of departing baggage units, split up into originating and transfer baggage

The model input provides information regarding:

- Flight schedule and all specific information per flight such as statistics of number of bags arriving and departing, driving time to and from assigned gate, initial arrival delay.
- Pier configuration: Lay-out, number and location of gates
- Locations of baggage sorter outlet, reclaim and transfer inlet.
- Distance matrix
- Number of available baggage vehicles as a function of time and specifications
- Number of available load teams as a function of time and specifications.
- Penalty factors used in scheduling algorithms
- Factors characterising the level of congestion as a function of time
- Run control information

**EXPERIMENTS AND RESULTS**

**Comparing the Concepts**

Table 2 shows the experimental set up resulting in 8 different concepts to be investigated. Concept 0 represents the current situation: short distances, baggage train, no dynamic scheduling. The objective of the simulation project is to compare the 8 concepts on the basis of logistic performance indicators and costs.

<table>
<thead>
<tr>
<th>Table 2: Eight Investigated Concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance to gates</td>
</tr>
<tr>
<td>Means of transport</td>
</tr>
<tr>
<td>Dynamic scheduling</td>
</tr>
<tr>
<td>Concept Nr</td>
</tr>
</tbody>
</table>

The logistic performance and service levels are represented by a number of indicators:

- Gate occupation
- Vehicle utilization
- Personnel and equipment
- Departure delay
- Transfer times
- Reclaim times

The performance indicators are combined in a weighted judgement. Because this is an arbitrary measure, the weighing factors are model-input and the sensitivity with respect to the factors can be investigated. One simulation run covers a period of 24 hours and is conducted with a real flight schedule. For every concept 10 runs with different random streams for the distributions are carried out and averaged. The standard deviation for the average delay is in the order of magnitude of 25%. All performance indicators are directly generated in the simulation experiments. Figure 4 shows a part of the animation screen of the model. Next some examples of model results will be shown.

*Gate occupation and vehicle utilization*
Figure 5. Gate-Occupation and Vehicle-Use as a Function of Time for Concept 3, with the New Baggage Truck on the D-pier using Dynamic Scheduling

Figure 5 shows the gate occupation and vehicle use as a function of time for concept 3. It can be seen that the gate occupation is rather low with the current time schedule. This is the consequence of the flight schedule used. The vehicle availability has been adapted to the expected work load of the system in order to reduce labour costs.

Delay due to baggage operations

Departure delay is a result of arrival delay and possible extra delay due to gate operations. Figure 6 shows the net departure “delay” of airplanes due to overrunning the schedule of the baggage operations. The net departure delay is corrected for initial delays at arrival of the flights. If the flight arrives too early the initial delay is set to zero. Mind that the net departure delay may be negative, or in other words it is possible to catch up an initial delay by handling the baggage faster than the standard handling time. It is concluded that the baggage truck performs better than the present train system.

Figure 6: Contribution of Baggage Operations to Flight Delay. Green bars show a “negative” delay, to be interpreted as “ready earlier than planned”, and red bars are positive delays.

Influence of number of personnel and equipment.

The model can be used to investigate the influence of the available people and equipment. This influence can for example be expressed in terms of departure time delay. Figure 7 shows the average total departure delay as a function of the equipment available. The concept numbers are indicated in the figures. The influence of dynamic scheduling is clear in the case of decreasing number of vehicles, if comparing for example concepts 0 and 1 for the baggage train and concepts 2 and 3 for the baggage truck.

Transfer and reclaim time

Transfer and reclaim times are the times needed to complete transportation of arriving baggage from the plane to the transfer inlet and the reclaim inlet respectively. The general trend is that the truck variants perform best.

Logistic Evaluation Using Weighing Factors

Figure 8 shows the weighted performances of all concepts. Although the weighing is arbitrary, the results indicate that the concepts 3 and 7 (truck with dynamic scheduling at pier D and J respectively), prevail over the train concepts. The difference between train and truck system is most pronounced in the J-pier case. The reason for that is the advantage of the higher velocity of the baggage truck and the longer distances to be covered. The improvement of dynamic scheduling is significant.

Figure 8: Weighted Logistic Performances of All Concepts

Costs

Costs are directly proportional to the number of employees and number of equipment used and to the length of working time and the distance travelled. The only uncertain factor is introduced by the costs attributable to the baggage truck that is only in the design state. An estimate of the cost of the baggage truck was made. The number of people needed as well as the number of vehicles needed is strongly related to the required punctuality of the baggage handling system as can be concluded from figure 7. In which the relationship between number of vehicles available and the average
departure delay is shown for all variants. Costs are therefore influenced by the required service level. In Figure 9 the costs of all concepts are shown. The conclusion is that for both D- and J-pier cases the costs of the truck concepts are down to more than 50% of the costs of the concepts using baggage trains.

![Relative Costs for All Concepts](image)

Figure 9: Relative Costs for All Concepts. The costs of the current operating procedure, concepts 0 and 4, are set to 100% for the D-pier and the J-pier respectively.

ADDITIONAL EXPERIMENTS

In the simulated cases the gate occupation has been low (fig. 5). In a number of additional experiments the load of the system is increased to determine the maximum pier-capacity. The results of the experiments will be mentioned briefly.

Short fixed turn around times

The cases described up till now are derived from real time schedules. Only a small percentage of the flights have short prescribed turn around times. In order to investigate the influence of fixed small turn around times the original flight schedule was simulated again assuming that all turn around times are fixed to 30 minutes. So the same planes are simulated but each with a restricted turn around time of 30 min. The departure delay times measured appear to increase because they are related to the planned time of departure. Peaks in the delay times appear to be much lower when dynamic scheduling is applied. Further it was observed that short turn around times induce lower gate occupation, thus increasing the capacity of the pier. This experiment automatically leads to the question what is the maximum pier capacity.

Determination of maximum capacity

To that end a fictitious very busy flight schedule was composed. All flights are assumed to be conducted with planes of the largest narrow-body category. The prescribed turn around time was set to 30 minutes and the number of equipment and personnel was set unlimited. It was concluded that only the concepts using the baggage truck perform satisfactorily. It appears that the maximum capacity of the pier is almost a factor 3 higher than the capacity needed for the current flight schedule.

The existing baggage operation system using trains of baggage carts is compared with a system equipped with a new designed baggage truck. Further a scheduling heuristic was applied to assign jobs to loading teams and transport vehicles dynamically. These two approaches have been applied to a nearby pier and a pier with a long distance to the baggage sorting centre. It is concluded that the logistic performance of the baggage truck is better than that of the baggage train system. The truck-concepts show a potential saving of more than 50% when compared with the baggage train concepts.

REFERENCES


SIMULATION MODEL OF CATERING SERVICE MANAGEMENT FOR AERIAL TRANSPORT

Francesco Costantino  
Giulio Di Gravio  
Massimo Tronci  
Department of Mechanical and Aeronautical Engineering  
University of Rome "La Sapienza"  
Via Eudossiana 18, 00184 - Rome, Italy  
E-mail: francesco.costantino@uniroma1.it

KEYWORDS  
Aerial transport, catering service, management strategy, simulation, service time

ABSTRACT  
Air transport today can increase its competitiveness focusing on improving secondary activities, such as on board and ground handling services. Among them, catering service (meal preparation and on board setting up), is now having more and more importance. These services are commonly outsourced to catering companies that have to respect first of all strict contractual requirements with the Air Companies and on the other hand many security, safety and hygienic bonds. The research has as analysis object an optimization of catering service management for aerial transport by a simulation model. The results obtained are carried out from a collaboration with a company working in Rome “Leonardo Da Vinci” international hub and were used to choose which are the best practices for the service, in particular for the response to last minute events.

INTRODUCTION  
Catering service for an airline is a critical element under different points of view: first of all because, as the primary process of the transportation companies (transfer of things and people from one to another geographic place) is always guaranteed, the customer choice on the market is based principally on the price and furthermore on all the secondary services that are offered. On second hand, one of the first requests of the air company’s customers is the punctuality of the acquired service, that depends not only on the transportation process modalities but mainly on everything that happens around the plane and on board, before taking off and after landing.

Even more, if among the support service we analyze catering service, we have to face with hygienic and health requirements that are very rigid in an environment that leaves few organizational margins: the spaces for food management are always limited, as the one for stocking, flight times can be long and the quality has always to research higher standards.

To understand the complexity of such a kind of service, we can consider that in the international hub where the research was developed, our company can employ more than 900 people to produce a maximum of about 25,000 meal a day. Furthermore, a single long range Boeing 747 needs 42,000 different articles to complete the meals and to make safer and more pleasant the journey. A slowing down or a failure in the execution of the setting up activity has consequences on the entire preparation process of the airplane because all the ground handling activities need a great coordination (passengers landing, vehicle cleaning, luggage management...).

The research singled out that the setting up of the plane is the critical activity for a catering service company because most of the delays depend on it. On these reasons the correct management of these kind of societies has to focus on every activity that composes the process:

- prepare on board equipment for the serving flight;
- check the conformity (quality and quantity) of the requested output;
- load on board equipment on catering truck;
- prepare the airplane and load the returning units on the same catering truck;
- come back of the team and unload the units.

About plane setting up, the main factors (defined as last minute events) that influence the performance level of the service are:

- meals supply;
- equipment supply;
- airplane change.

It’s common to effectuate a further supply of missing meals, because of a different number of passengers on the flight, or an equipment supply, if requested by the flight assistants or by the crew of the destination airport. Furthermore, accidental events as engine or trolley damage before boarding cause an airplane change: this generates a new definition of quantity and typology of the units to load, because not only the plane has changed but overall the entire configuration.

SIMULATION OBJECTIVES

A simulation model was created to analyze the existent reality. The scope was to evaluate the influence of the stochastic variables and the process behaviour with
their changing, to propose improvement in management strategies and control the results obtained. The defined objectives of the study were:

- create a model that could represent the setting up activity;
- measure the catering service performance in stochastic unlucky events;
- validate possible management solution for last minute events.

In particular, the simulation was used as a tool to verify both designing and management strategies. The analysis focused on:

- setting up time;
- logistic flows dimensioning;
- efficiency of the employed resources (setting up team, personnel on platform, catering truck, fridge catering truck);
- check of the adopted solution;
- global balance of the system and resources allocation.

THE SIMULATION MODEL

The construction of the simulation model was effectuated defining three incremental sceneries:

- representation of the core setting up process;
- representation of the core process with last minute events;
- management of last minute events for long range flights.

The simulation was focused on long range flight because for those trips it’s necessary a higher consumption of resources caused by the volume of the units to check and handle (about 130 items).

The first scene (fig.1) designs the process identifying the core operations effectuated both inside the meal production site and on ground. The flowchart was modeled to allow an implementation with Witness simulation software. The model is constituted of two parallel production lines with two different checking points depending on the items (meal trolley or other equipment). From these two checking points the parts are loaded on a catering truck in defined proportion and then transferred on board. To model setting up and checking activities Gaussian distributions were used while the transportation process parameters, related to the conveyor, are considered deterministic.

The second scene introduced some disturbing elements. The frequencies of these factors that influence the correct development of the service and the effect in caused delay on the process were measured after about 1500 long range flights monitored. Even if the efficiency of the catering company can contain the situation that decrease the performance of the process, these factors highly affect delivery times and maintenance of meals trolley (tab.1).

![Fig.1 – Scene 1](image)

<table>
<thead>
<tr>
<th>Tab. 1 – Frequencies and time of last minute events</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Last Minute Event</strong></td>
</tr>
<tr>
<td>No delay</td>
</tr>
<tr>
<td>Meal supply</td>
</tr>
<tr>
<td>Equipment supply</td>
</tr>
<tr>
<td>Meal and equipment supply</td>
</tr>
</tbody>
</table>

Last minute events were simulated by machines with cycle time equal to the one needed to fill the requested level.

Last analysis (third scene) is a model of a possible strategy in process management. The proposed solution is not to congest the loading bay, with consequent inefficiency of the checking and handling resources, using a buffer on ground whose task is to answer directly to the unexpected demand of the flight companies. In particular it was thought to introduce a smaller cooling truck (Fridge Truck – fig.2) used as stock for meal supply (68% of available space) and equipment (32%) as buffer. During the day, Fridge Truck has to come back to the production site about three or four times a shift to replenish its stock level.

For each simulation scene, the work in progress time of meals and equipment for the entire process was calculated.

372
When in the model disturbing factors are introduced, the behavior of the system, related to the increase of the activities, reacts with drastic changes in handling dynamics. In fact, in the second scene, it’s evident how the actions to follow the increased demand level due to last minute events create consistent delays, if totally managed in the production site. Again, the output analysis (fig.4) gives an average time for all the activities of about 222 minutes and a standard deviation of 25.71. In comparison with the first scene it’s evident that, even if guaranteeing successful process in 53.87% of flights, last minute events percentage increases service time of about 31% and the dispersion from the nominal conditions.

Fig. 2 – Scene II, Gaussian time distribution

The simulation study, at this point, examined a management strategy of catering service to answer to last minute events, having both an upper and a lower limit to define its performance efficiency, fixing the frequency of happening of further supply events as external to the model. The entire process in the third scene has a deviation of about 13% from nominal condition (against 31% of the second). In particular, output data (fig. 5) give an average time of 191 minutes and a standard deviation of 17.8 that can be considered as an effective improvement, especially for the variability of the process. Furthermore, the average waiting time of the units to board is reduced from 118 to 96 minute (decrease of 18%) and the standard deviation from 32.05 to 22.98 (decrease of 29%). The use of the fridge truck buffer directly on ground allows to answer just in time to every last minute demand coming from the air companies and to lighten the work on the production site (fig.6 and 7).

Fig. 3 – Scene I, Gaussian time distribution
IMPROVEMENT MODEL
As last step, after having described an as-is analysis, a study was done in different environmental conditions:
- increase and modification of last minute events frame (tab.2);
- use of fridge truck only for meal maintenance and supply;
- in site management of equipment supply and airplane change.

**Tab.2 – Last minute events frame in IV scene**

<table>
<thead>
<tr>
<th>Last Minute Event</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>No delay</td>
<td>45.10 %</td>
</tr>
<tr>
<td>Meal supply</td>
<td>39.07 %</td>
</tr>
<tr>
<td>Equipment supply</td>
<td>10.13 %</td>
</tr>
<tr>
<td>Airplane change</td>
<td>5.70 %</td>
</tr>
</tbody>
</table>

The feasibility of using a fridge truck exclusively for the meal supply was verified considering also the following parameters:
- maximum capacity of fridge truck (1500 meals);
- maximum number of contemporary medium and long range flight in peak hours (15 flights);
- probable estimation of requests for a single flight (22 passengers);

The results obtained in the forth scene (fig.8) of the simulation showed an improvement of the global time, as the entire process centres on an average value of about 183 minutes and a standard deviation of 17.15 minutes. The same improvement can be noticed in the waiting time distribution (average of 89 minutes, standard deviation of 20.86 – fig.9).
CONCLUSION
The simulation study had as objective an analysis of the management techniques for a catering service in setting up process. An incremental model of different sceneries permitted to compare the results of the different runs using graphic and statistical report elements available in the software. The upper and lower limits, obtained by simulating the core process and all the disturbing elements, showed that the proposed solutions make the model robust to the influence of stochastic variables due to balancing meal or equipment supply and to airplane change. In the end, with an hypothesis of a change in condition caused by demand oscillation, a management solution is presented without new investments in equipment or structures but just using in an efficient way the already available resources.

BIBLIOGRAPHY
VALIDATION
ROSSY: A TOOL FOR SYMBOLIC SIMULATION OF HYBRID SYSTEMS

Elke Tetzner, Wolfgang Lohmann and Günter Riedewald
Department of Electrical Engineering and Computer Science, University of Rostock,
18051 Rostock, Germany
E-mail: (tetzner|wlohmamn|gri)@informatik.uni-rostock.de

Abstract

Hybrid systems, a well-known modelling technique for embedded real-time systems, need to be proved for time- and safety-critical properties. We verify such properties by symbolic simulation, a kind of automatic verification. The symbolic simulation is executed in our tool Rossy, consisting of the modular declarative language MODEL-HS for description of hybrid systems and queries for properties, a translator from MODEL-HS to the constraint-logic language PrologIV and a symbolic simulator in PrologIV. The paper introduces the architecture of Rossy and hybrid systems. As a fundamental process for execution of symbolic simulation, description and translation of hybrid systems from MODEL-HS to PrologIV are explained in more detail. Properties are usually expressed in temporal logic. As a consequence, we give an idea of relationships between our queries and well-known temporal formalism.

1 INTRODUCTION

Hybrid systems are models based on a set of hybrid automata (Alur et al., 1993; Alur et al., 1995) for realistic and comprehensible description of concurrently running computer processes embedded in a physical environment. Continuous elements for the continuously progressing quantities of the environment together with discrete actions and reactions of the interacting computer system are combined in hybrid automata. The automata can be interpreted concerning two labeled transition systems (Henzinger, 1996): a timed transition system that abstracts continuous progress by transitions and a time-abstract transition system that additionally abstracts the duration of the progress. Such interpretation allows us to consider behaviour and properties of hybrid systems as a set of timed trajectories. Timed trajectories are sequences of events that are connected with the time of their occurrences, so called time stamps. Such a description of behaviour and properties makes the proof of properties using simulation possible. In our case, time stamps can be variables, i.e. symbolic values for an infinite range of numeric values. In this way, we execute a symbolic simulation (Riedewald, 1995; Riedewald & Urbina, 1995) for verification of temporal properties in hybrid systems which is strongly related to bounded (Biere et al., 1999; Biere et al., 2003) and constraint-based model checking (Delzano & Podelski, 1999; Delzano & Bultan, 2001). The approach has been already successfully explored with our cultivation system of biosensor cells (Tetzner et al., 2000) and examination rules for computer science courses at the University of Rostock (Lantsman et al., 2003).

We developed the tool Rossy (ROStock SYmbolic simulation) for proving properties with the components, seen in figure 1: i) MODEL-HS (Tetzner & Riedewald, 1999), a MODular DEclarative Language for Hybrid Systems and queries for properties, ii) a translator in Lapbot (Lämmel & Riedewald, 2001) for translation of hybrid systems and queries from MODEL-HS into PrologIV, a constraint-logic language and iii) a symbolic simulator in PrologIV. A model in MODEL-HS is based on hybrid automata. A query in MODEL-HS is divided into symbolic trajectories and three kinds of conditions connected to the trajectories. A precondition informs about initial values of parameters and variables. A commitment limits the length of proved trajectories for termination. Properties for continuous time and variables determine relationships between time stamps and continuously calculated variables. Both descriptions: i) the model of a system and ii) the query are translated to PrologIV, using our language processing tool box Lapbot. The model in MODEL-HS is split into a symbol table and a hybrid system. The symbol table is a technical instrument to achieve a hierarchy for identifiers and variables in the flat structure of the hybrid system. The query in PrologIV includes preconditions, commitments and properties. The separated trajectory as well as the hybrid system and the symbol table represent parameters for the query. A symbolic simulator, written in PrologIV, calculates conditions according to the query. Trajectories, which fail the conditions, do not meet the queried property.

In section 2, we introduce an example system as foundation for the next sections. Section 3 deals with the translation of descriptions in MODEL-HS to PrologIV in detail. The idea of relationship between our queries and temporal expressions is topic of section 4. Finally, summary and future work conclude the paper.
2 TRAFFIC LIGHTS

As an example we use a system of three traffic lights for pedestrians in figure 2. The main characteristics of the system are:

1. All three traffic lights have two visible states: 'red means stop' and 'green means go'.
2. Light 1 takes a change to green only if a pedestrian requires such a signal by pressing a button.
3. Light 2 and 3 change frequently concerning a given clock rate. Light 3 changes more often than light 2.
4. Light 1 must synchronize with light 2.

Figure 2: Crossing with traffic lights for pedestrians

In figures 3, 4 and 5 hybrid automata for the processes of the three traffic lights are depicted. Every automaton is connected with a local clock $X$ which is set to zero at initial transition. The periods of red and green can be found in locations where continuous time progress is described with differential equations and strictly limited by invariants, formulated as mathematical inequalities. In this way, locations represent an infinite range of states which are all calculated values of local clocks here. In the hybrid automaton for Traffic_Light1 we have integrated an auxiliary location $r.g$. The location models a necessary time progress to wait for a change to green after a requirement of a pedestrian. The locations are connected using transitions which can be triggered by events. Events are either actions, modelled with received signals for synchronization, or time conditions. An event can cause new events, modelled with sent signals, and a reset of clocks.

Figure 3: Traffic Light 1

Looking at figures 3, 4 and 5, all three automata are intended to begin with their red phase. The phase should also be the end location for one cycle of a process, marked with a double circle. While Traffic_Light1 has to wait for a requirement of a pedestrian, symbolized with sent signal ped_press, the other two lights can and must change from red to green and from green to red periodically: i) 'can' results from time conditions $X > 3$ at discrete transitions which mean after three time units a change from one location to another might be happen where a signal for information to the environment is emitted and clock $X$ is reset, ii) 'must' results from invariants $0 \leq X \leq 4$ of locations which mean a process might be remain in a location from zero to including four time units, but
Figure 4: Traffic Light 2

Figure 5: Traffic Light 3

after the time the location must be left. Here, efficient using
of powerful modelling with hybrid systems for such a simple
looking example is amazing. Practically, the time conditions
at discrete transitions specify time periods for stopping and
go of pedestrians and the invariants describe time periods
for stopping and going of vehicles.

If there is a requirement of a pedestrian to Traffic_Light1
then the process has to wait for the next red phase of light 2.
The waiting time is limited by the invariant of location r_g which
must be left after 7 time units. On the other hand, the pedes-
trian must wait at least one time unit according to the condi-
tion $X > 7$ at the discrete transition. By triggering the transi-
tion a signal turn_to_green is sent and clock $X$ is reset. Af-
after further 3 time units turn_to_red might be emitted where
clock $X$ is reset. The transition is triggered not only by a time
event but also by the optional signal green_light2. Optional
signals have to be accepted immediately by their occurrences
if time conditions are fulfilled. Otherwise, time conditions
are sufficient for triggering a transition. Here, three cases
were caused by the approach: i) green_light2 is received
within 3 time units then the signal may not be accepted, ii)
green_light2 occurs after 3 time units then the transition can
be triggered and iii) at all times after 3 time units and before
the occurrence of green_light2 the transition might be trig-
gered. By the approach additional transitions can be omitted.
For instance, the transition with optional signal green_light2
represents two transitions: one with the time condition $X > 3$
and the other one with time condition $X > 3$ together with an
additionally received signal green_light2.

3 TRANSLATION OF HYBRID
SYSTEMS

Descriptions of hybrid systems in MODEL-HS require the
translation of hybrid systems from MODEL-HS to Prolog IV
as main precondition for execution and symbolic simulation
of such systems. The language MODEL-HS supports special
elements and structures for specification of hybrid systems
and grouping for a good overview as well as reusing of parts
and entire systems in other hybrid systems. In this way, we
close a gap between comfortable execution and monolithic,
hard to read descriptions of hybrid systems in Prolog IV.

Hybrid systems in MODEL-HS are devided into automata
and blocks in analogy to SDL (Ellsberger et al., 1997). Hy-
brid automata are the basic elements, which synchronize us-
ing send and received signals and communicate with shared
variables. Blocks are combinations of automata for modular-
ization and structuring a set of hybrid automata. Blocks can
be nested. The block, which includes all other blocks and
automata, is called hybrid system. On next page, the automa-
ton of Traffic_Light1 can be seen. In a parameter list, input
and output signals are specified, followed from a declaration
part for clock $X$ and locations red, r_g, green. In the body
of Traffic_Light1 the initialisation, termination and transition
part can be found. If there is no signal or condition for ini-
tialisation, we use the default value true. Interesting is the
while-structure for describing continuous progress in a loca-
tion. This is a powerful expression. The element while is fol-
lowed by common invariants for several locations. The part
in leads to locations, which can be connected with specific
invariants, only valid for the locations. Here, we addition-
ally specify condition $X = < 7$ for r_g and condition $X = < 4$
for green. The do-expression informs about the progress
of continuous clocks and variables in an algebraic notation.
A set of in-do-combinations can belong to the same while-
expression. Discrete transitions give information about leav-
ing and reaching locations, sent and received signals, condi-
tions and assignments of variables during a transition. The
interpretation of discrete transitions for Traffic_Light1 corre-
sponds to the description of section 2.

Components for modelling of computer programs are well
formulated and executed with rules of logic programming
and otherwise, continuous and discrete changes of environ-
mental quantities together with conditions on the quanti-
ties are specified adequately and calculated efficiently with
constraints. Consequently, PrologIV as constraint-logic lan-
guage supports an execution mechanism for our symbolic
simulation of hybrid systems (Riedewald & Urbina, 1995).
Every automaton has to be translated into PrologIV with four
kinds of rules as displayed on next page in the second box.
The rules init represent initialization parts of automata. Only
clock $X$ is set to zero for Traffic_Light1 to reach its initial
location red. In the continuous rules, progress and invari-
ants of clocks and variables of locations are described with
constraints in the body of the rules. Three constraints are de-
scribed for location r_g of Traffic_Light1. As result of math-
ematical interpretation, constraints can be ordered arbitrarily.
Traffic_Light1 {in ped_press, red_light2, green_light2;  
out turn_to_green, turn_to_red;  

declaration  
clock X;  
location red, g, green;  

begin  
initialization  
location red;  
if true -> X:=0;  

termination red;  

transitions  
continuous  
while 0 <= X  
in red, g(X<=7), green(X<=4)  
do X:=X+1;  

discrete  
leave red reach g  
if receive[ped_press] -> X:=0;  
leave g reach green  
if receive[red_light2], 1 < X  
-> send[turn_to_green], X:=0;  
leave green reach red  
if [receive[green_light2]], X > 3  
-> send[turn_to_red], X<=0;  

end Traffic_Light1;  

init[receive([]),send([]),’Traffic_Light1’,’red’,<Param, [’<X’, Xold, Xnew, 0>], ContinVar,DiscVar,ControlVar>);  

continuous(’Traffic_Light1’,’r_g’,<Param, [<’X’, Xold, Xnew, Xold>], ContinVar,DiscVar,ControlVar>);  

Timegap := 0 <= Xnew, Xnew <<= 7, Xnew = Xold + Timegap.  

discrete[receive([’red_light2’]),send([’turn_to_green’]),’Traffic_Light1’,’r_g’,  
’green’,<Param, [<’X’, Xold, Xnew, 0>], ContinVar,DiscVar,ControlVar>);  

<= 1 < Xold.  

synchronous([’red_light2’,’turn_to_green’],’Traffic_Light2’,’Traffic_Light1’),  
[’Traffic_Light1’,’Env’], Timegap, g_Space, r_Space, g_Space, r_Space);  

continuous(’Traffic_Light2’,’green’, g_Space, r_Space);  

continuous(’Traffic_Light1’,’r_g’, r_Space, Timegap);  

discrete[send[red_light2],’Traffic_Light2’,’green’,’red’, g_Space];  

discrete[receive([’red_light2’]),send([’turn_to_green’]),’Traffic_Light1’,’r_g’,’green’, g_Space].  

Here, Timegap is a variable which includes the lapse of time from entering location r_g up to execution of the continuous-rule now. The discrete-rules describe all parts connected with a discrete transition. Only conditions, as for discrete transition between r_g and green, form the body of the rules. Assignments, as X := 0, are formulated in the parameter-list of the head, here with <’X’, Xold, Xnew, 0>. Synchronization between automata is specified with synchronous-rules where all other rules are called in correct order, changing continuous and discrete progress permanently. Here, Env describes an automaton for the environment, which is abstract without any further details. In the context, it is sufficient to say that all tiny displayed variables are for calculation or buffering results for next rules only.

4 HYBRID AUTOMATA AND TEMPORAL PROPERTIES

Looking at the example, generally queries for liveness and safety of pedestrians and vehicles should be answered where every group has enough time for going and has to stop in defined time intervals. In this section, we outline the relationship between expressions of temporal logics and structure of our queries because transformation from the expressions into the queries is really worth for implementation to give a user the chance to specify properties in a well-known, comfortable and precise mathematical way. We are able to prove properties such as following informal proposition:

At some time t in the future if a pedestrian requires green at Traffic_Light1 then the other two lights should have red signals soon. The red signal of light 2 should precede the red signal of light 3. Both red signals should occur earlier than turn_to_red at Traffic_Light1. The proposition should be true before 12 time units from beginning of execution.

In the temporal logic TPTL (Alur & Henzinger, 1990) the proposition could be formalized as follows:

$$\text{Qt.ped_press -> (¬red_light2Uu.red_light3 ∧ ¬v.turn_to_red ∧ u < v < 12)}$$

The meaning is that there a time instant t connected to the eventually-operator ∨ exists where a true proposition ped_press implies proposition red_light3 at time u and proposition turn_to_red at time v soon. Proposition red_light2 must be true at least one time before proposition red_light3 is met. This is expressed with the until-Operator U. Time u should be less than time v and all time instants should be less than 12 time units. Mapping the propositions to 4-tuples of (Signal, Sender, Receiver, Symbolic Time):

(‘red_light2’, ‘Traffic_c_Light2’, ‘Traffic_c_Light1’, Tau_r2)
(‘turn_to_red’, ‘Traffic_c_Light1’, ‘Env’, Tau_tg)
we can formulate a corresponding query:

\[
\text{search\_trajectory} \{ \text{Head\_Traj,'red\_light2', 'Traff\_c\_Light2', 'Traff\_c\_Light1', 'Traff\_c\_Light3', 'Traff\_c\_Light1', 'Tail\_Traj} \}:
\]
\[
\text{in} \ (\text{Head\_Traj}, \{ \text{'ped\_press', 'Env', 'Traff\_c\_Light1', 'Tail\_p} \} ,
\text{not\_in} \ (\text{Head\_Traj}, \{ \text{'red\_light3', 'Traff\_c\_Light3', 'Traff\_c\_Light1', 'Tail\_r3} \},
\text{in} \ (\text{Tail\_Traj}, \{ \text{'red\_light3', 'Traff\_c\_Light3', 'Traff\_c\_Light1', 'Tail\_r3},
\text{('turn\_to\_red', 'Traff\_c\_Light1', 'Env', 'Tail\_tg} \},
\text{Ta} \text{u}_{r3} < \text{Tau\_tg} < 12.
\]

This means, calculate all trajectories, which consist of three parts: a list of tuples at the beginning, a tuple for event \text{red\_light2} and a list of tuples at the end. Event \text{ped\_press} must be given in the list of beginning tuples but \text{red\_light3} must not. Events \text{red\_light3} and \text{turn\_to\_red} must occur in the tail of trajectories before 12 time units, first \text{red\_light3} and after this \text{turn\_to\_red} according to our last rule \text{in} and the constraint \text{Tau\_r3 < Tau\_tg} < 12. The foundation of the transformation is the general relation between sequences of propositional expressions and expressions of temporal logic as used in (Audemard et al. , 2002). In this sense, our challenge is to get a suitable classification for queries in Rossy concerning temporal expressions. Symbolic simulation replies queries in Rossy. Results of symbolic simulation are constraints on which trajectories meet queried properties. The constraints cover complete ranges of values for continuous clocks and variables. Consequently, we can execute an automatic verification with the approach which gives opportunities for detailed discussions.

5 CONCLUDING REMARKS

The tool Rossy permits an execution of hybrid systems with symbolic simulation as automatic verification in a pleasant way. There, MODEL-HS provides as modular language useful problem-oriented concepts for specification of hybrid systems and queries for properties. Otherwise, PrologIV enables us to apply a powerful combination of mathematically calculated constraints and logic description for execution of symbolic simulation. The paper contributes to the translation from MODEL-HS to PrologIV for closing the gap between comfortable execution and complex, inconvenient descriptions of hybrid systems in Prolog IV. Successful practical applications of the approach can be find with examples of our cultivation system (Tetzner et al., 2000) and planning problems for studying (Lantsman et al., 2003). Also the example of the paper is a little fragment of a high potential traffic engineering problem. In the future, concepts should be explored to give a general connection between queries for properties described with trajectories and expressions in temporal logic. Beyond, MODEL-HS should be extended to parametrized automata and blocks. Here, we could describe similar components with same locations and transition structures but different expressions for invariants, actions for progress, conditions and assignments for discrete transitions. We intend to integrate our visual language VYSMO (Tetzner et al., 2001) into Rossy. In conclusion, the symbolic simulator in PrologIV must be improved for more efficiency.

REFERENCES


AUTOMATING PART OF THE MODEL VERIFICATION PROCESS

Rob Walker & Habtom Mebrahtu
Department of Design and Engineering
Anglia Polytechnic University
Bishop Hall Lane, Chelmsford
Essex CM1 1SQ, United Kingdom
r.w.walker@apu.ac.uk
h.mebrahtu@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
Verification, validation, cusum methods, simulation

Abstract
The modeller frequently overlooks the verification of the simulation models of a manufacturing line as defined by Balci (1994). Part of the verification process should be to change the random number stream generator seed values and determine if this has caused a significant change in the model results. This paper demonstrates the need for verification by showing results that are significantly different and the actions that should be taken to overcome the problem. The paper also demonstrates an automatic process for verifying the model and returning the confidence limits. Thus the analyst will be able to report the expected output of a line between these limits. This should prevent the misunderstanding by management that a manufacturing line will always output exactly the same number of parts per week, whatever the circumstances. For example, the analyst will report that the expected output will be between x and y parts per week with 95% confidence.

Model Validation and Consistency Check
The model shows a typical manufacturing line of a company in the Chelmsford area that has been disguised to ensure company confidentiality. The line manufactures nozzles and consists of 4 operations and operates a single shift system for 37 hours per week (2220 minutes per week). Table 1 shows the detail of the processes in the line.

| Operation | Cycle time (mins) | Breakdown | Duration of breakdown | No. of machine
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mark</td>
<td>2.75</td>
<td>Negative exponential decay mean = 5600</td>
<td>See table 2</td>
<td>1</td>
</tr>
<tr>
<td>Grind</td>
<td>8.6</td>
<td>Negative exponential decay mean = 2000</td>
<td>See table 3</td>
<td>3</td>
</tr>
<tr>
<td>Hone _1</td>
<td>7.8</td>
<td>Negative exponential decay mean = 2000</td>
<td>See table 3</td>
<td>2</td>
</tr>
<tr>
<td>Hone _2</td>
<td>7.0</td>
<td>Negative exponential decay mean = 2000</td>
<td>See table 3</td>
<td>2</td>
</tr>
</tbody>
</table>

The simulation model for the existing system was constructed in Witness using the above cycle times and breakdown data. In the Witness model the breakdowns occur at random triggered by the pseudo random numbers generated by Witness. This means that events are not truly random and unless the model is changed, the result of each run of the Witness model will always return the same answer. To comply with the Balci (1994) verification process for a simulation model the random number stream generated by the computer should be changed. A program has been written to automate this random number stream changing to allow a statistical analysis to be carried out on the results to verify the consistency of the result. If the result is not statistically stable then the model must be reviewed. A common error that could result in an unstable model is breakdowns of long duration, that occur relatively infrequently. The effect on the results when a breakdown occurs would be to reduce the expected output considerably, but when it does not occur then output predicted would be inflated. This could obviously lead to incorrect conclusions being drawn from the model and unrealistically optimistic or pessimistic results being recorded and inappropriate action taken. Whether optimistic or pessimistic results are returned could depend solely on the random number stream allocated by the modeler.

Table 2

<table>
<thead>
<tr>
<th>Duration of breakdown (mins)</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 150</td>
<td>2</td>
</tr>
<tr>
<td>150 - 200</td>
<td>10</td>
</tr>
<tr>
<td>200 - 300</td>
<td>25</td>
</tr>
<tr>
<td>300 - 350</td>
<td>28</td>
</tr>
<tr>
<td>350 - 400</td>
<td>9</td>
</tr>
<tr>
<td>400 - 600</td>
<td>2</td>
</tr>
</tbody>
</table>
To verify the consistency (stability) of the model a program has been written in Visual Basic within Excel that checks the model by conducting 25 runs changing all the random number streams for each run. This check for consistency is a part of sensitivity analysis as recommended by Balci (1994). The cusum chart shows the ‘unexpected effects (that) may reveal invalidity’ Balci (1994).

Outputs from 25 runs were recorded and a Cusum was constructed as shown in Figure 2 based on one week’s output (2220 minutes). The mask used is a C2 semi-parabolic mask as defined in BS 5703 part 3 (1981). As can be seen from the graph in Figure 2 one data point was outside the upper mask, indicating a lack of consistency. Figure 3 shows another run in this case based on 4 weeks output and the data point is just inside the mask showing that the model is valid. Another run based on 12 weeks production indicates an improved consistency.

The purpose of the stability check is to ensure that the model is valid and that changing the random number stream will not significantly affect the model. If it did then the model would be unstable and the model results cannot be expected to give a solution that would be realistic.

It was decided to use a control chart to monitor stability as it gives a graphical representation of the data so that trends and inconsistencies can be found easily and most engineers in a manufacturing environment will be familiar with control charts.

A cusum chart was selected to assess stability because it is recognized as the best SPC chart for detecting small changes in performance that would not be detected by a Shewhart chart (Owen, 1989). The benefit of using the cusum chart is the long-term development potential to monitor and control the simulation that enables the simulation to automatically:

- Detect if the model is stable and valid as described above;
- Initiates a change in the model that may result in a significant improvement in the system performance;
- Identifies when a change does not make a significant improvement and can be discarded;
- Indicates when the optimisation process is complete within the predefined parameters.

When the optimisation process is complete a rerun of the program will ensure that the model is still valid.

To make the data generated during the above analysis the confidence limits can be easily calculated for the 95% confidence limits. The 95% confidence limit has been selected as this is the confidence limits recommended by Grove & Davis (1992) for industrial experiments. The confidence limits for the above example for weekly output based on the t distribution are given in Table 4.

<table>
<thead>
<tr>
<th>Run length simulated weeks</th>
<th>Mean</th>
<th>Standard deviation</th>
<th>Lower 95% confidence limit</th>
<th>Upper 95% confidence limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>406.5</td>
<td>17.51</td>
<td>399</td>
<td>414</td>
</tr>
<tr>
<td>4</td>
<td>461.7</td>
<td>10.64</td>
<td>453</td>
<td>471</td>
</tr>
<tr>
<td>12</td>
<td>473.5</td>
<td>5.81</td>
<td>465</td>
<td>482</td>
</tr>
</tbody>
</table>

**Cusum Method**

The cusum chart in table 5 requires two pieces of information to initiate the system they are the target value and the sample standard deviation. We suggest that the mean of the initial results is the target value and the sample standard deviation is the s value used to calculate the mask limits. A minimum of 25 readings should be used to calculate these variables.

The calculation of the cusum is a two stage process; the first stage is to calculate the difference between the target value and the data point (x-T) and then to add this result to the previous cusum. The first cusum is simply the first x-T value. Using the data in table 5 as an example of the cusum method the calculations are as follows. The mean of the output quantities is 406.52 and this is used as the target value (T) in the calculations. The first simulated output is 412 parts (x), thus x-T value is 412-406.52 = 5.48 and this is also the first cusum value. The second simulated output is 419 parts and the x-T is 419-406.52 = 12.48, this is added to the last cusum to give the current cusum of 5.48+12.48 = 17.96. The third x-T is 418-4.06.52 = 11.48 and the cusum is 17.96+11.48 = 29.44. This process is repeated until all the simulated output values are processed. The normal rules of arithmetic apply when dealing with negative numbers.

A cusum chart in essence is about slope: if the cusum points are approximately horizontal then the readings are near the target value and the process is probably under control. If the slope is upward then the readings are above target and the process may be out of control. If the readings are less than target, the slope will be downward.

A mask is used to determine if the slope is too great for the process to be considered under control. The mask is prepared to suit the process and the scale of the chart and is drawn on acetate or ‘Perspex’ for durability. The focus of the mask is placed on the last reading; all the previous readings should fall within the arms of the mask; if they do, then the process is under control.

The semi-parabolic mask that we recommend is equivalent to a 2.65σ action limit and 1.65σ warning limit (BS 5703,
1981). Other mask designs are available; see BS 5703 (1981) for further details.

The value of s is used to draw the mask that is the sample standard deviation (Coleman, 1996) of the x-T values for the results from the initial results.

Figure 1. Typical mask design

Figure 2 Cusum chart based on one simulated week

Conclusions

At present this system is not fully developed, but we believe that it has the potential to prevent action being taken on invalid models that may result in recommendations being made that are inappropriate and could result in either unnecessary expenditure or the under capitalisation of a production line.

Figure 3 Cusum chart based on 4 simulated weeks

Figure 4 Cusum chart based on 12 simulated weeks

REFERENCES

Balci O, Validation, verification, and testing techniques throughout the life cycle of a simulation study, Annals of Operations Research 53 (1994) 121-173

BS 5703 Part 3 (1981) Data analysis and quality control using cusum techniques

Coleman, Greenfield, Jones & Puzey, The Pocket Statistician, Arnold 1996

Grove D M & Davis T Engineering Quality & Experiment Design, Longman 1992

Owen M, SPC and continuous improvement, IFS, 1989
Figure 5 Witness model
<table>
<thead>
<tr>
<th>Run No</th>
<th>Nozzle A</th>
<th>4</th>
<th>14</th>
<th>19</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Total Parts X</td>
<td>42</td>
<td>41</td>
<td>49</td>
<td>50</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
</tr>
<tr>
<td>Cusum</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Upper mask</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Lower mask</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Snells</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Target</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>5</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Sigma</td>
<td>1</td>
<td>7</td>
<td>5</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td>3</td>
<td>9</td>
<td>5</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>Critical factors</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Cusum</td>
<td>5</td>
<td>4</td>
<td>7</td>
<td>6</td>
<td>9</td>
<td>4</td>
<td>7</td>
<td>6</td>
<td>9</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>Total Parts Y</td>
<td>42</td>
<td>41</td>
<td>49</td>
<td>50</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
<td>49</td>
</tr>
<tr>
<td>Cusum</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Upper mask</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Lower mask</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Snells</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Target</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>5</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Sigma</td>
<td>1</td>
<td>7</td>
<td>5</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td>3</td>
<td>9</td>
<td>5</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>Critical factors</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Cusum</td>
<td>5</td>
<td>4</td>
<td>7</td>
<td>6</td>
<td>9</td>
<td>4</td>
<td>7</td>
<td>6</td>
<td>9</td>
<td>4</td>
<td>7</td>
</tr>
</tbody>
</table>
TECHNIQUES FOR THE SPREADSHEET DEVELOPMENT OF PROFESSIONAL MODELS

Hervé Thiriez
HEC School of Management
78350 Jouy FRANCE
E-mail: thiriez@hec.fr

KEYWORDS
Spreadsheet-based models, Decision support systems, Model design, Interactive simulation.

ABSTRACT
There are a number of advantages to using spreadsheets in the development of professional models: spreadsheets are present on all computers, they are very interactive, they allow for rapid and cheap developments, and there is little human resistance to the use of spreadsheet-based models. The limits of using spreadsheets for professional models are size limits, i.e. 256 columns and 65,536 rows (this is less of a restriction however), calculation speed, calculation errors or problems, the difficulty of model auditing...

Most of these limits can be dealt with in an efficient manner, which will be adequate in most situations. This paper comments on these limits and shows how most of the problems derived from these limits may be solved.

INTRODUCTION
There are a number of advantages to using spreadsheets in the development of professional models: spreadsheets are present on all computers, they are very interactive, they allow for fast and cheap developments, and there is little human resistance to the use of spreadsheet-based models.

The limits of using spreadsheets for professional models are size limits, i.e. currently 256 columns and 65,536 rows (the number or rows is less of a problem), calculation speed, calculation errors or problems, difficulty of model auditing...

We will comment on these limits of spreadsheet-based model development and show how some of the problems related to these limits may be solved in practice.

ROW AND COLUMN LIMITS
The 256-column limit is probably the major problem I ever had to solve when using spreadsheets for the development of professional models.

Let us assume you create a model for the time simulation of a waiting line, and the time unit is 30 seconds. With 256 columns, you can only simulate a 2-hour process.

At some time, I was developing a planning model for a rolling mill. The base task of the mill was 6 minutes long, which meant that a model dedicating a column to each 6-minute slot could not represent more than one day of activity. The problem was that the planners also needed to have a 5-day view of the situation, where they did not necessarily want to see all the details.

We see below the model for the mill (Laminage, in French, in row 2) where each of the following rows represents one of the heating ovens. Other ovens are hidden by TT and RW.
The solution found for the model was to have a switch, activated by a click on the sand-clock in cell A1, which transforms the 6’ time interval into a 30’ time interval. A macro then redraws the schedule accordingly. Of course, a detailed analysis of the planning was meaningless in this representation, as several events could happen in the same 30’ slot and, of course, only one of them would have been visible in the cell. However, with this feature, it became possible to have a 5-day view of the jobs. Below, we can see such a view with the TT ovens open (the opening or the closing of the TT ovens is achieved by a click in the TT button in A30). Going back and forth between the 6’ and the 30’ views, it was easy to have a good understanding of the planned period.

| A     | H     | I     | J     | K     | L     | M     | N     | O     | P     | Q     | R     | S     | T     | U     | Y     |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Lamina |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| Arrêt lamina |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| FAL1   |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| FAL2   |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| FAL3   |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| Cellule 2 |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| Cellule 3 |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| Cellule 5 |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| Cellule 6 |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| FSM2   |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| TT 1   | S=75096,6798 |       |       |       |       |       |       |       |       |       |       |       |       |       |
| TT 3   |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| TT 5   |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| TT 6   | S=2115,S=679899,679899,5657 |       |       |       |       |       |       |       |       |       |       |       |       |       |

The limit with the number of rows is not so stringent and, when a large database has to be used, it is often more efficient to handle the complete database with a tool such as Access and to extract databases that Excel still manages properly.

With adequate VBA developments, it is however possible to overcome this limit by having the code handle bases split between several sheets of the same workbook. And, since VBA code runs much faster than Excel database functions, specific macros often make it possible to do efficient analyses on such multi-sheet bases.

**CALCULATION SPEED AND PROBLEMS**

Complex spreadsheet models often show calculations times which are too long, and sometimes not even acceptable. There are several techniques for solving this speed problem.

The first technique is to clean the model, i.e. to get rid of any redundant or obsolete formula.

A formula is redundant if the same result is calculated in several cells: in such a case, it is often better to put in a cell the intermediate result which will be used by several other cells.

A formula is obsolete when the data used by the formula are data which will not be modified in the future. This often happens with sheets containing historical data, e.g. one column per month over three years. When you are quite sure some “old” columns will not be modified any more, you can select them, copy them, and “paste special” their values.

One should always remember that Excel calculates from top to bottom and, in each row, from left to right. Another technique for speeding up calculations is, knowing this, to organize your sheets in such a way that all your formulas are evaluated according to this calculation order.

The last and most efficient way to speed up calculations is to have macros put the formulas in the cells, calculate, replace by the values, go to the next group of cells and do the same operation again, … Of course, you gain tremendously in speed, but at a cost: your models becomes difficult, at best, or impossible, at worst, to modify or audit properly.

I did this on a model I had designed for the simulation of a toll booth system. When the model was initially developed with all its formulas, it took 45 minutes to simulate two hours of traffic. After I had developed macros to put the formulas and calculate the results, the simulation time has decreased to less than 2 minutes.

There were even larger savings, in proportion, when I had – in another model – macros which replaced Excel’s database functions for very large databases.

Calculations problems are something else. There are two major such problems: Excel refuses to calculate even though you use the [F9] key, and Excel really gets wrong results. When Excel becomes unable to calculate with a simple [F9], you often solve the problem by using the [Ctrl]-[Alt]-[F9] combination. If this still does not work, you may resort to an astute “Replace All” of “=” by “=!”

Excel getting wrong results is altogether another problem. This is due to the fact that Excel only has a precision of around 14 digits, but this is the initial precision. The more operations you do, the smaller the precision. This means that, when some data are subject to many operations, it is wise – whenever possible – to use at some point a technique which will restore the original precision. Numerous such techniques have been developed, which may be found in the literature.

**MODEL AUDITING AND MODEL VALIDATION**

What good is a model if you cannot understand how it works or if you cannot validate its results?

By nature, spreadsheet models are not easy to audit. Model auditability apparently has never been a Microsoft priority, which can easily be verified by this small experiment: if you right-click on an Excel toolbar, you get a list of available toolbars, but the “Audit” toolbar is not to be seen in that list.

In order to see it, you have to use the “Customize” command and look for the bar!
Models can be made more auditable quite simply by using names: a formula such as “= Product cost + Margin” is far better than “=B6+K8”. A model is also easier to audit if all the blocks are well isolated: data blocks, intermediate calculations, results, …

Last, there are also auditing tools, some of them Excel add-ins, which help you better understand a model and automatically notice apparent inconsistencies.

Model validation is a quite more difficult issue, too often overlooked in professional developments. This problem was solved in an interesting fashion for a car-traffic simulation model I created for Aéroparts de Paris (AdP), in order to simulate the car traffic in the 20-square km airport area.

During our initial analysis of the complete layout, we selected all the points on the map where we wanted to have car counters. These counters measured car speeds and numbers, on a minute per minute basis. The counters were installed in the many selected positions and took minute per minute measures during all sorts of full days: standard day, weekend, vacation time, …

In this fashion, many Excel models (one per day, with a row per minute and a column per point) were created, through direct links with the measure apparatus. On a common agreement with AdP, we were provided with half of these data files, AdP keeping the remainder for the validation phase.

I strongly believe in bottom-up development, which I always use in practice. Therefore, I began by developing initially a relatively straightforward model. I then used the first few Excel data files in order to see how the model behaved: this allowed me to enrich the model so that I could obtain more realistic results.

When this was done, I opened additional data files and checked my model with these new data, which again led me to add new features to my model and enrich its formulas.

Adding data files progressively, and checking the model at each step, I enriched it until it behaved in a satisfactory fashion with all the data files in my possession.

I then gave my model to AdP, and they checked it with the data files they had kept. My model was validated by AdP because the results it predicted were coherent with what their data files had recorded.

When they agreed that the model behavior was satisfactory, they ordered the model for a second of the five traffic blocks. I am currently finishing the development of the model for the third block, but the basic model formulas did not need to be modified since the end of the development of the first block.

CONCLUSION

In this paper, we showed that some problems have to be solved if one is to develop professional models with a spreadsheet. Luckily, most of these problems have at least partial solutions, most of them not demanding to resort to Visual Basic.

One of the major advantages for using a spreadsheet in the development of professional models is that the development time and cost are much smaller than with the competitive approaches.

Whenever I find myself in a tender offer competition, my spreadsheet-based solution is 3 to 5 times cheaper and faster than traditional solutions with big modeling packages, or based on development languages.

When a model is built faster, it is not only cheaper, it is also much more realistic. The shorter the development time, the less discrepancy there is between the problem which was initially analyzed and the problem at the time the model has become operational: the model is therefore more realistic and thus more efficient. It is a complete win-win situation: faster, cheaper and better.

REFERENCES


BIOGRAPHY

HERVE THIRIEZ has an engineer’s diploma in Applied Mathematics from IMAG (Grenoble, 1966) and a Ph.D in Operations Research from MIT (1969), where he taught for one year. He then returned to France for his military service and became a Professor at the HEC School of Management. He has been an Operations Research consultant for 35 years and is the CEO of Logma, a consulting company. He also published 25 books and more than 300 papers on modelling and micro-computing.
AN INTEGRATED SIMULATION AND VERIFICATION ENVIRONMENT IN SUPPORT OF THE SYSTEM LIFE CYCLE

Mike Rennie, Vicente Fernandez and Augusto Caramagno
Deimos Space, S.L.
Ronda de Poniente, 19
Edificio Fiteni VI, 2-2
28760 Tres Cantos (Madrid)
Spain
E-mail: mike.rennie@deimos-space.com

KEYWORDS
Software life cycle, Simulation, Verification, Real-time test bench.

ABSTRACT
This paper describes the simulation and verification framework under which the activities of the DEIMOS Real-Time Laboratory have been initiated. This framework is intended to support the development, verification and validation of flight systems and software, according to a combined systems and software engineering life cycle. It aims at optimising the collaboration between the systems engineers and software developers, and at maximising the re-use of software and tools between the phases of a project, as well as between different projects.

INTRODUCTION
DEIMOS has undertaken the definition and the effective implementation of an integrated simulation and verification environment (Rennie 2003). This effort is based on a process model that supports both the early system engineering activities, as well as the development and testing of the onboard software. Nowadays, the point at which the system engineering finishes, and the software engineering process takes over, has become less obvious, and in reality there is a fair degree of overlap. This is especially true when a significant amount of code is generated automatically from the models developed by the system engineers. This is most apparent when dealing with GNC algorithms, or communications protocols.

The outcome of this effort is a versatile framework architecture, whose reconfiguration capabilities and components are described in the following sections. The main components of the architecture, namely the Functional Engineering Simulator (FES) and the Real-Time Test Bench (RTTB) are described to a certain detail. To end up, an illustrative study case is presented, depicting the facility developed by DEIMOS in the frame of the Independent Verification & Validation of the GOCE onboard software.

BACKGROUND
Immersed in an increasingly competitive market, space systems design methodologies are rapidly evolving towards innovative concepts aiming at the reduction of costs and time. Early approaches to more efficient spacecraft design were based on collaborative and concurrent design technologies (Sarsfield 1998).

In the last years, the Virtual Spacecraft concept is emerging as a new paradigm of simulation facility for aerospace design applications, relying on the availability of advanced modelling and simulation software tools (Benac 2003). The virtual spacecraft consists of a set of functional and geometrical models corresponding to all the relevant subsystems so that the engineering team can perform and iterate the design, prototyping, integration and verification processes, before the system is physically built, as illustrated in Figure 1.

![Figure 1: The Virtual Spacecraft within the system entire Life Cycle (Benac 2003)](image)

A direct consequence of the availability of virtual models of the spacecraft during the beginning of the design is that the definition and the analysis of the Assembly, Integration and Test (AIT) process can be performed at an early stage of the project, which implies significant cost savings (Braghin 2003).

To mention some recent implementations of this concept, we cite the ATPE Simulator, developed in the frame of ESA’s Aeroassist Technology for Planetary Exploration (ATPE) project, which consists in a rapid prototyping environment for GNC algorithm development and validation (Da Costa 2003). Also, the Model-based Development and Verification Environment (MDVE) goes further in the definition of an integrated framework supporting from the early development stages to the satellite Assembly Integration and Verification (AIV) activities (Eickhoff 2003).

COMBINED SYSTEM AND SOFTWARE ENGINEERING LIFE CYCLE MODEL
In order to clarify the roles and responsibilities of the
system and software engineers during the production of the onboard software, it is convenient to adopt a new approach to the traditional waterfall life cycle for SW engineering, to better take into account the collaborative effort between system and software engineers. A life cycle model adapted in this way is the Combined System and Software Engineering Process Model, as shown in Figure 2. The legend at the bottom of the figure indicates who is responsible for each activity, either software engineers or system modelling engineers, or whether the responsibility is shared among them. This process model is based loosely on the traditional V-model, with the specification, design and implementation activities generally on the left-hand side and the test activates on the right.

![Combined System and Software Engineering Process Model](image)

**Figure 2: Combined System and Software Engineering Process Model**

In the **SW Requirements Partitioning** activity, we decide which parts of the overall software will be coded manually, or will be modelled and subsequently coded automatically. Subsequently two sets of corresponding requirement specifications are produced, i.e. one for the parts that are to be coded manually by the software engineers, and another for the parts that are to be coded automatically from corresponding models created by the system engineers. The process then splits into two parallel life cycles accordingly, but with several points of rendezvous and coordination.

The aim of the **Top-Level Architectural Design, Partitioning and Interfaces Definition** activity is to ensure close cooperation and coordination between the software and system engineers, so that the overall OBSW architecture and its internal interfaces are consistent and specified by mutual understanding and agreement.

The “pure” software engineering life cycle continues with further decomposition of the top-level architecture, to derive the complete architectural (physical) model of the software, with “placeholders” in the design for the autocode components that will be integrated into the software later on. The development continues with the detailed design of the “pure” software modules, and subsequently the manual coding.

Meanwhile, after the **Model requirements for Autocoded Components** activity, the engineering model development life cycle proceeds with the (selection and) customisation of the modelling tools, in the **Modelling Tool Customisation** activity. This is followed by the **Model Design and Development** activity, in which the models are produced, using the selected modelling tools. It is to be expected during the course of this activity, as the models are developed and refined, that there may be a need to modify the interfaces with the manually coded components. Therefore, the process foresees the need to coordinate
between the system and software engineers at this point. This is shown in the process diagram by the broken arrows representing inputs from the system engineers, i.e. the Model Design and Development activity, to the software engineers (the two activities that deal with the software architecture).

The developed models are verified using the simulation and model verification capabilities of the simulation framework (Figure 3). They are then coded automatically. The verification of the quality of all of the code is performed at this point, irrespective of whether it is coded manually or automatically.

Actual testing of the source code begins with unit tests. Unit tests may be skipped only when the code has been generated automatically by a code generator that has been certified to the required level. However, if the code generator is not certified to at least the required level, then both the verification of the models and the unit tests are planned and executed.

Continuing up the right-hand side of the process, we come to the activity Integration of Code, in which the parallel "pure" software and system engineering model life cycles rejoin. At this point, since we are dealing with the overall software, the responsibility for the integration is given to the software engineers. If the interfaces were defined carefully and completely during the earlier architectural design phase, this integration should go smoothly. Nevertheless, it would not be a surprise to discover problems in the detail of the interfaces, and once more this is a potential point of friction between the software and system engineers, that must be managed with care.

Following the integration of the entire code, the software engineers proceed with the (traditional) Integration Tests activity. Subsequently, the software and system engineers collaborate in the Validation Tests activity, to validate the software with respect to the complete set of requirements. Finally, the software and system engineers collaborate, with the customer, in the Acceptance Tests activity.

The process model also takes account of the iterative nature of the development. This is shown by the dash-dot-dash arrows. The evaluation of the quality of the automatically generated code may lead to the need to update the models, or even to customise the modelling tools in order to correct a deficiency in the code generation templates. The integration of the automatically generated code with the manually written code may uncover problems of compatibility, thus leading also to customisation of the tools.

SIMULATION AND VERIFICATION FRAMEWORK ARCHITECTURE

At present, the DEIMOS simulation and verification framework is composed of two major elements: the Functional Engineering Simulator (FES) and the Real-Time Test Bench (RTTB), as shown in Figure 3. Basically, while the FES supports the first stages of the life cycle, from system specification and design to prototyping, the RTTB is dedicated to the later verification and validation phases.

The capability to support different phases of the project life cycle is a primary design driver of this framework. Such a versatile concept is able to serve several functionalities by changing its configuration. For this purpose, adequate interfaces have been defined so that the components of the architecture can be arranged as:

- **Simulation Facility**: corresponds to the FES in stand-alone mode, when real-time performance is not a requirement.
- **Software Verification & Validation Facility**: when the OBSW is generated by autocoding, this configuration comprises the FES, the Onboard Computer (OBC) or a proper emulator for executing the binary image, and the Real-Time Simulator (RTS), which simulates the immediate environment of the OBC, including I/O devices.
- **Real-Time Test Bench**: in this case, a real-time performance configuration is required, including the RTS, the OBC, the Ground Segment Simulator (GSS) for TC/TM emulation and, in most cases, Hardware-in-the-loop (HIL) components for hardware testing.

![Figure 3: Simulation Framework Architecture](image)

**FUNCTIONAL ENGINEERING SIMULATOR**

At an early stage in the system development, when the emphasis is on the system, rather than onboard software, engineering aspects, the analysis of the technical challenges and requirements is supported by the development and utilisation of a Functional Engineering Simulator (FES).

![Figure 4: Functional Engineering Simulator Architecture](image)

This section provides a top-level description of the FES architecture in terms of control and data flow between the first level macroscopic modules of the tool. These models are illustrated in Figure 4.
The **System Definition** module allows the user to build-up the reference mission for simulation. It includes the definition of the mission arcs and environment, of the desired topology and of requirements and constraints affecting the mission.

The **Simulator Engine** performs the simulation process monitoring and control and handles the configuration of the simulation parameters. The simulator engine uses the outputs from the Simulation Core to monitor the simulation.

The centre of the architecture is the **Simulation Core** supported by the simulator engine. Basically, the Simulation Core contains the dynamical models of the spacecraft’s subsystems, including models of the GNC functions, functional models of sensors and actuators, and the Dynamics, Kinematics and Environment (DKE) models, e.g. orbital and attitude dynamics and kinematics, and environmental perturbations (Figure 5).

![Figure 5: A generic scheme of Simulation Core](image)

**REAL-TIME TEST BENCH**

At the heart of our Real-Time Laboratory is the Real-Time Test Bench (RTTB), whose basic layout and main elements are illustrated in Figure 6:

- The **On-board Computer** (OBC) Simulator is devoted to execute the OBSW code as directly generated by the autocoding tool from the GNC models.

- The **Real-time Simulator** (RTS) runs the code corresponding to the dynamics, kinematics and the environment (DKE) of the launch vehicle, actuators and sensors, in a dedicated real-time computer. The executed code is generated through auto-coding from FES dynamical models.

- The **Ground-Segment Simulator** (GSS) emulates the TM/TC front-end so that telecommand packets can be sent to the OBSW and the generated telemetry data can be monitored.

- The **Hardware-in-the-loop** (HIL) component usually refers to the target processor, although other hardware devices connected to the OBC could also be included for specific purposes.

The application of combined rapid-prototyping and autocoding technologies (Figure 2) to the development of the RTTB significantly reduces the necessary effort in terms of cost and time. In fact, the design of both FES and RTTB has been carefully studied in order to take advantage of using auto-coding tools. Figure 7 illustrates the autocoding process, which migrates the GNC models into OBSW to be executed in the OBC, and the actuator, sensor and DKE models into real-time software to run in the RTS.

![Figure 6: The Real-Time Test Bench](image)

The target (ERC32) processor resides in the OBC, namely a VME development computer (Tharsys). It serves as the hardware-in-the-loop on which the onboard software (OBSW) is executed.

The OBC communicates, currently via Ethernet, with the Ground Segment Simulator (GSS), a SCOS2000-in-a-box hosted on a Linux PC, thus emulating the TM/TC front-end interface, which permits Telemetry & Telecommand packet exchanges with the OBSW to be performed.
On the other side, the RTS hosts the simulation models of the environment. They consist of models of the actuators and sensors, as well as models of the dynamics, kinematics and space environment, but also of models of the environment of the target computer, e.g. I/O and communication devices. These models are able to talk with the target computer, at present via serial line, but this is soon to be extended to cover Mil-1553 bus, and there are plans to also include Spacewire links and CAN bus.

To support the rapid prototyping of embedded application software (ASW), the RTTB includes a library of device drivers that can be used by the ASW through a clean application programming interfaces (APIs), as well as reusable application software components that serve as building blocks for constructing the typical onboard software functions such as the services defined in ESA’s Packet Utilisation Standards. The components are designed to be integrated with the most commonly used embedded run-time systems (RAVEN, ORK, RTEMS, etc.).

![Image of the Auto-coding Process]

**Figure 7: The Auto-coding Process**

**GOCE ISV, A STUDY CASE**

The SW-only simulation artefacts of the RT-Lab form the nucleus of the CDMU simulation facility being developed by DEIMOS in the scope of the Independent Software Verification & Validation of the GOCE onboard software. The ISVV covers the platform basic SW, data handling application SW, drag-free attitude control application SW, AOCs sensors and actuators embedded software). Particularly close attention must be paid to testing the Platform onboard SW executing in the CDMU. For that purpose, DEIMOS is developing a specific high-fidelity simulator of the CDMU and its interfaces with the rest of the intelligent devices (Figure 8).

The fidelity of the overall simulation depends on what the user puts into the models that are connected to the ERC32 emulation software tool (TSIM), but it is perfectly possible to produce a simulation that faithfully reproduces the I/O data and control flows and protocols, with accurate simulated timings, at bit level. For GOCE, the environment that has to be modelled, and integrated into the simulation based on TSIM, includes the behaviour, as observable by the onboard software, and logical control and data interfaces with the:

- immediate environment of the processor, i.e. the other components inside the CDMU, such as the TC and TM modules, the reconfiguration module
- immediate environment outside the CDMU, such as the spacecraft discrete TM/TC and low-level interfaces with the sensors and actuators, and the data handling buses (Mil1553B)
- intelligent remote units used by the Platform OBS, such as the AOC sensors and actuators, the Star Tracker, the Gradimenter, and the SSTI.

![Image of the GOCE ISV facility]

**Figure 8: Simulation Core of the GOCE ISV facility**

**CONCLUSIONS**

DEIMOS Integrated Simulation and Verification Environment is proving its worth in terms of reducing the cost and risk typically associated with the development of onboard software, in particular for mission- and safety-critical applications. It provides the benefits of a cheap but reliable alternative to validating the software on the real system. It also goes a long way towards bridging the traditional divide between system and software engineering. The feasibility of implementing system functionality in onboard software is examined and analysed at an earlier stage in the overall system life-cycle, aided by the ability to use autocode techniques and simulation (optionally with representative HW-in-the-loop) as a means of rapid prototyping. This further reduces the risk in the development of the onboard software.

**REFERENCES**


**PERFORMANCE EVALUATION OF MULTI-SERVER QUEUES WITH STATION AND SERVER VACATIONS**

Nawel Gharbi and Malika Ioualalen  
Department of Computer Science,  
Département d'informatique - USTHB,  
BP. 32 – EL ALIA, Bab Ezzouar 16111,  
ALGER- ALGERIE  
E-mail : nawel_gharbi@hotmail.com

**KEYWORDS**  
Multi-server Queueing systems, Server vacation, Station vacation, Generalized Stochastic Petri nets, Performance Analysis.

**ABSTRACT**

This paper deals with finite-population, single and multi-server queueing systems with two classes of vacation mechanisms: station vacation and server vacation. In the first one, all the servers take vacation simultaneously whenever the system becomes empty, and they also return to the system at the same time, i.e., station vacation is group vacation for all servers. For the second class of vacation, each server takes its own vacation whenever it completes a service and finds no customers waiting in the queue.  

We show a method for modeling these queueing systems using Generalized Stochastic Petri Nets. We develop the general form of the underlying CTMCs, and we give two algorithms for computing the infinitesimal generator $Q$ for systems with $s$ servers ($s \geq 1$). Hence, several quantitative performance measures can be obtained for queueing systems with station or server vacation.

1. **INTRODUCTION**

Queueing systems in which the server takes a vacation from a primary task either to attend to one or more secondary tasks or to rest, arise as models of many computer, communication, production and other stochastic systems. For example, in polling systems, such as the token-passing local networks, the server attends to a group of queues in a cyclic manner. From the point of view of a particular queue, the server goes on vacation each time it ceases to serve that queue. So, the server working on the secondary customers is equivalent to the server taking a vacation and not being available to the primary customers during this period. Similarly, processors in computer and communication systems do considerable testing and maintenance besides doing their primary functions (as processing telephone calls, processing interactive and batch jobs, receiving and transmitting data, etc). The testing and maintenance periods which are mainly done to preserve the sanity of the system and to provide high reliability, may be regarded as server vacations. Hence, customers (messages for example) who arrive while the server is on break (vacation) have to wait until he is back. The machine breakdowns which may occur randomly, independent of the queue status, in manufacturing systems for example, can also be regarded as server vacations. Thus, there is a natural interest in the study of queueing systems with server vacations. This class of queues has been extensively studied by several researchers. The readers are referred to the surveys of (Doshi 1986; Doshi 1990; Teghem 1986), the monograph of (Takagi 1991), as well as the references therein.

However, in almost all these papers, it is assumed that there is one single server in the system, and that the population is infinite. In this article, we consider queueing systems with finite population and multiple servers ($s \geq 1$).

Two vacation mechanisms are introduced: station vacation and server vacation. For the first one, all the servers take vacations simultaneously. That is, whenever the system is empty, all the servers leave the system for a vacation, and return to the system simultaneously when the vacation is completed. So, station vacation is a group vacation of all servers. This occurs when a system consists of several interconnected machines that are inseparable, or when all the machines are run by a single operator. In this case, if the system (or the operator who runs the system) is used for a secondary task when it becomes empty (or available), all the servers (the operator) will then be utilized to perform a secondary task. During this amount of time, the system is unavailable to further arrivals, and this is equivalent to taking a station vacation. The second vacation mechanism, server vacation, is encountered even more often in practice. In this case, each server is an independent working unit, and it can take its own vacation upon completing serving a customer and finding no customers waiting. This phenomenon occurs, for instance, in post offices where, when observing an empty queue, a clerk goes for other types of work (sorting, distributing, etc).

In this paper, we show a method for modeling and analyzing finite-population multi-server queueing systems with station and server vacations, using generalized stochastic Petri nets (GSPNs).

One of the advantages of using GSPNs to model queueing systems with vacations, is that we can incorporate features that may be somewhat difficult to model by the more conventional methods. For example, if the server is subject to breakdown in addition to his normal vacation, the two kinds of vacations can easily be modeled by GSPNs in a hierarchical manner.
On the other hand, GSPNs are a useful tool for the development of very compact and easy to understand models of parallel systems with interacting concurrent components. This approach allows to obtain several benefits both for the qualitative and the quantitative analysis of these queueing systems. In particular, it offers the possibility of using results, methods and tools developed within the GSPNs framework, to obtain performance indices of multi-server queues with station or server vacations.

The paper is structured as follows: First, the basics of GSPNs are reviewed. In section 3, we give the GSPN describing multi-server queueing systems with server and station vacation mechanisms. Next, we develop the general form of the underlying continuous time Markov chains (CTMCs), and the algorithms for computing the infinitesimal generator Q, for a general number of servers s and a general finite-population capacity N. Finally, we give a conclusion.

2. AN OVERVIEW OF GENERALIZED STOCHASTIC PETRI NETS

In the past decade, GSPNs have received much attention from researchers in the performance and reliability arena, and have been extensively used for analytical modeling in the context of independability, performance and performability of computer, communication, manufacturing and aerospace systems. Many structural and stochastic extensions have been proposed in recent years to increase their modeling power, and their capability to handle large systems. GSPNs are an important graphical and mathematical model appropriate for describing and analyzing systems that are characterized as being concurrent, asynchronous, distributed, non-deterministic and/or stochastic.

A GSPN is a directed bipartite graph that consists of two nodes called places (drawn as circles), and transitions that are partitioned into two different classes: timed transitions and immediate transitions (Ajmone et al. 1995).

- Timed transitions describe the execution of time consuming activities. In this paper, we consider markovian timed transitions, that fire after an exponentially distributed firing time.
- Immediate transitions have priority over timed transitions and fire in zero time once they are enabled. This class of transitions was added to Stochastic Petri nets because modeling real systems often involves the presence of activities or actions, whose duration is short, or even negligible, with respect to the time scale of the problem. Hence, it is desirable to associate an exponentially distributed firing time only with those transitions which are believed to have the largest impact on the system operation.

In the graphical representation of GSPNs, timed transitions are indicated by means of rectangles, and immediate transitions are drawn as thin bars.

Formally, a GSPN is a six-tuple \((P, T, I(\cdot), O(\cdot), W(\cdot), M_0)\) where:

- \(P = \{P_1, P_2, \ldots , P_n\}\) is the set of places,
- \(T = \{t_1, t_2, \ldots , t_m\}\) is the set of transitions,
- \(I(\cdot) : P \times T \rightarrow \mathbb{N}\) is the input function which provides the multiplicities of the input arcs from places to transitions,
- \(O(\cdot) : T \times P \rightarrow \mathbb{N}\) is the output function which provides the multiplicities of the output arcs from transitions to places,
- \(M_0 : P \rightarrow \mathbb{N}\) is the initial marking which describes the initial state of the system,
- \(W(\cdot) : T \rightarrow \mathbb{R}^+\) is a function that associates rates of negative exponential distribution to timed transitions and weights to immediate transitions.

The system state is described by means of markings. The marking in general, is a mapping from P to \(\mathbb{N}\) which gives the number of tokens in each place after each transition firing.

Markings enabling no immediate transitions are called tangible markings. In this case, any timed transition can fire next (application of race policy commonly).

Marking enabling only immediate transitions are passed through in zero time and are called vanishing markings. In this case, only the enabled immediate transitions are allowed to fire because the lowest level is reserved for timed transitions.

Since the process spends zero time in the vanishing markings, they don’t contribute to the dynamic behavior of the system. So, a procedure is envisaged to eliminate them from the reachability graph which contains all the reachable markings from initial one (Ajmone et al. 1995).

Given a tangible marking \(M_0\) directly reachable from a vanishing marking \(M_n\), by the firing of an immediate transition, and \(S\) the set of tangible markings from which \(M_n\) is directly reachable. The vanishing marking \(M_n\) can be merged with \(M_0\) and so, eliminated by introducing arcs directly connecting \(M_0\) to \(M_n, \forall M_n \in S\).

This elimination of vanishing markings results in a reduced reachability graph which is isomorphic to a continuous time Markov chain (CTCM) due to the memoryless property of exponential distributions.

The states of the CTMC are the markings in the reduced reachability graph, and the state transition rates are the exponential firing rates of timed transitions in the GSPN. The solution of this CTMC at steady-state is the probability distribution over the set of the states. The steady-state distribution \(\pi = (\pi_1, \pi_2, \pi_3, \ldots)\) with:

\[
\pi_i = \lim_{t \to \infty} \text{Prob} \{ X(t) = i \} \quad (1)
\]

is computed as the solution of the linear system of equations:

\[
\pi Q = 0 \quad (2)
\]

\[\sum \pi_i = 1\]
where Q is the generator matrix (transition matrix). Having the steady-state probabilities vector \( \pi \), we can easily compute several performance measures of the studied system.

3. GSPN MODELS OF QUEUEING SYSTEMS WITH VACATION

We consider queueing systems in which customers arrive at the system according to Poisson process with rate \( \lambda \). There are \( s \) parallel servers (\( s \geq 1 \)), and the service times of the customers are independent and identically distributed exponential random variables with rate \( \mu \). The vacation times of all servers also are assumed to be independent and identically exponentially distributed with rate \( \theta \).

Any arriving customer who finds all servers busy upon arrival waits in the queue until a server will be idle. Every customer requires to be served by one and only one server, and leaves the system once the service is completed.

We consider multiple vacations policy (Doshi 1986; Doshi 1990). That is, the server or station, upon returning from a vacation shall leave immediately for another one if the system is empty at that moment. The process continues until the servers return and find any customer waiting.

The exhaustive service discipline (Doshi 1986; Doshi 1990) is assumed here. That is, upon completing a vacation, the server(s) return(s) to the system and start(s) to serve customers, if any, till the system becomes empty.

3.1. Queues with Server Vacation

This model is used for describing many practical problems where servers take individual vacations. This means, whenever a server completes service and there are no more customers waiting in the queue, it takes a vacation independently of others servers state.

A typical example is encountered in the post office. When a clerk completes services and finds not customers waiting, he might go to work on a secondary task, say sorting letters. This is what we refer to as the server vacation model.

Figure 1 shows the GSPN model describing the above system.

- The place \( P_0 \) contains \( N \) tokens, which represents the condition that none of the \( N \) customers has arrived for service. \( N \) is the population capacity.
- The place \( P_6 \) contains the customers waiting for service.
- The place \( P_s \) contains the number of customers in service.
- The place \( P_d \) represents the idle (available) servers.
- The place \( P_c \) contains the servers that are on vacation.

Currently, the marking (initial marking) of the net is \( M_0 = \{M(P_0), M(P_6), M(P_s), M(P_), M(P_d)\} = \{N,0,0,0,0\} \) which implies that none of the customers has arrived for service and that the \( S \) servers are idle.

- When the transition \( t_4 \) fires, one token is taken from \( P_a \) and deposited in \( P_b \). The firing of \( t_4 \) indicates the arrival of a customer. This firing is said to be "marking dependent"; that is, the firing rate of \( t_4 \) depends on the number of tokens in \( P_a \). Thus, if there are \( k \) tokens in \( P_a \), then the firing rate is \( \lambda_k \), and if there is only one token, the firing rate is \( \lambda_1 \). The condition of marking dependent firing is represented by the symbol \# placed next to the transition. The timed transitions \( t_5 \) and \( t_6 \) are also "marking dependent".
- The immediate transition \( X \) is enabled when the place \( P_8 \) contains at least 1 token, which represents a waiting customer, and the place \( P_d \) at least one idle server. Hence, one token is taken from \( P_b \) and from \( P_d \) and deposited in \( P_8 \). This token represents a customer in service.
- The input arc from \( P_b \) to the immediate transition \( Y \) is an inhibitor arc. The firing of \( Y \) represents the event that a server is commencing a vacation since there is no customer left to be served.
- When the timed transition \( t_7 \) fires, which represents the end of the vacation time, one token is deposited in \( P_d \) which represents the fact that a server is returned to the system.
- When the timed transition \( t_8 \) fires, one token is deposited in \( P_a \), which represents the condition that the customer has returned to be idle or thinking state and one token is deposited in \( P_d \), which represents the condition that this server is ready to serve another customer.

It is assumed that \( S \leq N \); that is, the servers number cannot exceed the population size of the system.

Note: For our model, tokens still accounting in the input places of transition, so much his firing not finished.
Considering all possible values for the population size $N$ and the servers number $s$, one obtains the CTMC for this model as follows:

![CTMC Diagram]

Where $M_{ix}$ is the marking $i$ in the level $x$, describing the system state at a given moment.

The general form of these CTMC states are as follows:

$$M_{ix} = [N-i, i-x, x, 0, S-x]$$

where $0 \leq x \leq S$ and $x \leq i \leq N$

The infinitesimal generator of this CTMC is a $IK \times IK$ matrix $Q$ where $IK$, the markings number of all CTMC equals $\Sigma(N-i+2)$ for $1 \leq i \leq S+1$

This transition matrix $Q$ can be constructed as follows:

$$Q((i,x),(j,y)) = \begin{cases} 
\lambda[(i,x),(j,y)] & \text{if } (i,x) \neq (j,y) \\
\Sigma \lambda[(i,x),(k,z)] & \text{if } (i,x) = (j,y) \\
(0,0) & \text{if } (k,z) = (i,x) \\
(0,0) & \text{if } (k,z) \neq (i,x)
\end{cases}$$

where $\lambda[(i,x),(j,y)]$ is the transition rate from marking $M(i,x)$ to marking $M(j,y)$ that can be calculated by applying the following algorithm:

**Algorithm 1**:  
**Begin**  
For $x := 0$ To $S$  
Do For $i := x$ To $N-1$ Do  
Begin  
$\lambda[(i,x),(i+1,x)] := (N-i)\lambda$;  
If $x > 0$ Then $\lambda[(i+1,x),(i,x)] := x \mu$;  
End;  
End;  
End.

For $i := 1$ To $S$  
Do For $x := 0$ To $i-1$  
Do $\lambda[(i,x),(i,x+1)] := (s-x) \theta$;  
For $i := S+1$ To $N$  
Do For $x := 0$ To $S-1$ Do $\lambda[(i,x),(i,x+1)] := (s-x) \theta$;  
For $x := 0$ To $S-1$ Do $\lambda[(x+1,x+1),(x,x)] := (x+1) \mu$;  
End;  
End;  
End.

Once the infinitesimal generator $Q$ is obtained, the vector of steady-state probabilities $\pi = (\pi_1, \pi_2, \pi_3, ...)$ can be computed by solving the linear system of equations $\pi Q = \pi$ with the normalization condition $\Sigma \pi_i = 1$

Having the steady-state probabilities vector $\pi$, we can easily compute several performance measures of the studied system.

3.2. Queues with Station Vacation

In this model, as soon as the system is empty (all servers become idle), the station takes a vacation. As one may expect, this situation appears to be more complicated that the previous one. In fact, it is more simple, because all servers take a vacation simultaneously and return to the system at the same time also.

Hence, the GSPN modeling this system with multiple station vacations, is the same model as the one given in Figure 1 in which the multiplicity of the arc connecting the place $P_0$ to transition $Y$ and transition $t_i$ to place $P_i$ equals $S$.
(rather than 1), because the S servers of the station take a vacation together. So, if the place Pd contains s idle servers, the immediate transition Y fires, which represents the begin of the station vacation time. At the end of this period (after a mean delay equals 1/θ), s tokens corresponding to the s servers will be deposited in Pd. On the other hand, the transition t, is not "marking dependent".

Considering all possible values for the population size N and the servers number s, the CTMC for the model with station vacation is given below:

![CTMC of Multi-Server Queues with Station Vacation](image)

Figure 3: The CTMC of Multi-Server Queues with Station Vacation

Where:

\[
\begin{align*}
M_{0,i} &= [N-i, i, 0, 0, 1], 0 \leq i \leq N \\
M_{1,i} &= [N-i, 0, 0, i, 0], 1 \leq i \leq S \\
M_{2,i} &= [N-i, i-S, 0, 0, 0], S+1 \leq i \leq N
\end{align*}
\]  

(5)

The infinitesimal generator of the CTMC of Fig. 3 is a IK*IK matrix Q, where IK=2N+1.

The components of Q are given by:

\[
Q[(i,x),(j,y)] = \begin{cases} 
\lambda[(i,x),(j,y)] & (i,x)\neq(j,y) \\
(N,1) & (i,x)\Rightarrow(j,y) \\
-\sum (k,z) & (i,x)\Rightarrow(j,y) \\
(k,z) & (i,x)
\end{cases} \]

(6)

The transition rates can be calculated by applying the following algorithm:

**Algorithm 2:**

**Begin**

For x := 0 To 1

Do For i := x To N-1

Do \(\lambda[(i,x),(i+1,x)] := (N-i)\lambda;\)

For i := 2 To S Do \(\lambda[(i,i),(i-1,i)] := i \mu;\)

For i := s+1 To N Do \(\lambda[(i,i),(i-1,i)] := s \mu;\)

For i := 1 To N Do \(\lambda[(i,0),(i,1)] := \theta;\)

**End.**

3.3 Single Server Queues with Vacation

The GSPN modeling the single server queue with station vacation is given in Figure 1 with S=1. Hence, the CTMC of the model is the same as the CTMC of Figure 3 where:

\[
\begin{align*}
M_{0,0} &= [N-1, 0, 0, 1], 0 \leq i \leq N \\
M_{1,1} &= [N-i, i-1, 1, 0, 0], 1 \leq i \leq N
\end{align*}
\]  

(7)

In fact, the two models with server vacation and station vacation give the same results for S=1. So, the models coincide since there is only one server in the whole station. Hence, for calculating transition matrix we can use one of the two algorithms given above.

4. CONCLUSION

In this paper, we proposed a technique that allows to represent finite-population, single or multi-server queueing systems with server and station vacations using GSPNs.

The novelty of the investigation is the multiplicity of the servers, and also the introduction of server and station vacation mechanisms, which make the system rather complicated.

The flexibility of GSPN modeling approach allowed a simple construction of detailed and compact models for these queueing systems. We have also developed the general form of the underlying CTMCs, and two algorithms for computing the infinitesimal generator Q, for a general value of the servers number and a general finite-population capacity. Hence, several quantitative performance measures can be derived.

Finally, many vacation queueing problems and their solution can be simplified using the GSPN modeling approach with all the results developed within this framework.

REFERENCES


HUMAN BEHAVIOUR MODELLING
ANALYSIS AND MODELLING OF MULTIUSER REALTIME INTERNET APPLICATIONS

Qili Zhou, Peter Astheimer, Colin J. Miller
IC CAVE and School of Computing and Advanced Technologies
University of Abertay Dundee
Dundee, DD1 1HG, United Kingdom
E-mail: cotqz, cotpa, mctejm @abertay.ac.uk

KEYWORDS
Modeling, Network traffic analysis, Multiplayer games

ABSTRACT

Overall Internet traffic is caused by many Internet applications active at the same time. Designing and dimensioning networks is problematic for some highly delay-sensitive Internet applications such as multiplayer games as the amount of traffic and response times of applications are difficult to predict. We investigated the potential of traffic modeling as a means to forecast network quality using an example of a popular multiplayer online game. We recorded network traffic of several Quake sessions, decoded communication metrics from captured files, obtained statistical models for the traffic and gauged discrepancies between empirical and analytic data.

1. INTRODUCTION

The rapid growth of Internet in both applications and users is urgently demanding researches on Internet traffic. The Internet traffic researches in 1990s have noticed simultaneous traffic can be very bursty (packets arrive in clumps) and are difficult to predict. Simultaneous traffic are known to have long-range dependence (LRD). LRD traffic means the autocorrelations are positive and exhibits fractal or self-similar scaling behaviour. Self-similarity traffic seems to look the same in the large (minute, hour) as in the small (second, millisecond). At every time scale ranging from milliseconds to minutes and hours, bursts consist of bursty sub-periods separated by even less bursty sub-periods. Because of the discovery of LRD traffic, Poisson arrivals and other tools of predicting call-blocking probabilities for telephony network are no more applicable to Internet traffic.

Studies on traffic of single Internet applications have been attracting many researchers. Network traffic were analyzed and modelled, suggestions of improving traffic behaviours were put forward. Delay-sensitive Internet applications have attracted more researchers than other applications. The general purpose of traffic researches of single Internet applications is investigating traffic characteristics and offering theoretical or statistical models. Characteristics of traffic metrics such as connections, bit rate, users behaviours round the time, packet count, packet size, arrival and inter-arrival time, correlations were investigated. Among the metrics, interarrival time and size are fundamental, their statistical statistic features were aimed in many reference.

2. MULTIUSER REALTIME INTERNET APPLICATIONS

A number of multiuser internet applications require a certain network quality to function satisfactorily, e.g. internet telephony, internet video and multiplayer games. Many games are more delay-sensitive than most other Internet applications. Game players have to make decisions in fractions of second in fierce battlefield situations. For example, the difference between 50 and 150 ms of delay can determine who wins or loses a game, but such difference is sometimes not noticed by users of IP phone call. Game suppliers and players long for better network communication. In order to improve game traffic and communication, it is important to investigate the characteristics of game packet communication.

By analysing game traffic, Wu-chang Feng et al found that game traffic provides significant challenges to current network infrastructure, because current routers are designed for bulk data transfers with large packets (packet size assumption expects average sizes are often between 125-250 bytes), but game packets are smaller. Routers must be designed with enough capacity to quickly route game traffic bursts without delay [Feng et al. 02]. This illustrates game traffic are worthy analysing and there is some potential way to improve game traffic behaviour.

Jeong-Soo Han et al studied delay patterns of voice traffic of end-to-end users on Voice over Internet Protocol (VoIP) network. It is extremely difficult to keep traffic delay and jitter to a minimum and assure good speech quality without controlling the burst rate of all traffic served [Han et al. 02].

According to Sarah Joyce, games do impact the throughput of TCP on a heavily loaded link. Any type of UDP traffic will limit the amount of TCP traffic that moves through the network. Games affect TCP in much the same way as VoIP. The underlying UDP protocol behaves in much the same manner regardless of the application that is running it [Joyce 00].

This tells us, controlling the traffic burst is important to VoIP. Because both VoIP and most multiplayer games share the support of UDP protocol, controlling the burst is also important to games, so game packet communications should be fully understood.

3. RELATED WORK ON MODELING
The researches on traffic of multiplayer online games have yielded analytic models about packet behaviours. Michael Borella’s work is probably the earliest to investigate packet-level models of multiplayer game traffic. He chose game Quake I and Quake II and gained statistical models of both packet size and interarrival time. Client packet size is well-modelled as deterministic, server packet size is extreme distribution. Both server and client interarrival times are well modelled with extreme distributions.[Borella 00]

Sarah Joyce et al explored game traffic of both Quake and Unreal Tournament. About packet size: they got the ranges of majority packet. About interarrival time: Quake client has two high peaks at 14 and 27 ms, and two low peaks at 40 and 53 ms, Quake server peaks at 14 and 26 ms. Tournament client peaks at 25 ms, server peaks at 60 and 240 ms. [Joyce 00]

Bangun and Dutkiewicz investigated game Star Siege Tribes. Client packet size is exponential distribution; interarrival time is a split distribution composing of two sections, both sections are exponential distributions. [Bangun et al. 00]

Dave LaPonte et al investigated traffic of both game Starcraft and Counter-Strike. For Starcraft packet size: 70% packets are at around 132-byte and the second peak is at 120-122 bytes. For Counter-Strike packet size, the mean value, together with the standard deviation are obtained. [Claypool et al. 03]

Johannes Färber characterized the traffic patterns of game Counter-Strike. Packet size and server interarrival time are the extreme distribution, client interarrival time is deterministic distribution. The parameters of the extreme distribution of server packet size vary, due to different number of clients. [Färber. 02] [Färber. 03]

Wu-chang Feng et al analysed traffic of Counter-Strike. Client packet size has an extremely narrow distribution centered around the mean size of 40 bytes, server packet size has a much wider distribution around a significantly larger mean. [Feng et al. 02]

4. QUAKE III

However, different references have slightly different conclusions in traffic modelling. For example, Quake interarrival time is extreme distribution [Borella 00] or has only peaks [Joyce 00]. Server packet size is extreme distribution [Borella 00] or has irregular pattern [Joyce 00]. FPS (First Person Shooter) games are highly delay sensitive. The general objective of FPS games is to explore a common virtual world and kill as many of the other players as possible. Most FPS games are based on either Quake engine or the Unreal engine, and game “Quake III Arena” is very fast [Färber. 02]. However in references, Quake III have not been analyzed. As to Quake II: server traffic to all clients were modelled aggregately instead of separately; some client has different distribution of interarrival time from the rest clients in the same experiment [Borella 00]. Several reference on Internet application analysed interarrival time (some were restricted by experiment conditions). If game traffic is focused, inter-send time is more accurate because interarrival time contains the distortions of queuing mechanisms. Several reference failed to measure discrepancy between analytic and empirical data (data are large and classic goodness-of-fit cannot apply easily).

5. EXPERIMENTS AND DATA COLLECTION

Like many other games, Quake uses UDP protocol for communication. A client communication cycle consists of reading a server packet, processing it, rendering players’ screen situations, sampling input device, transmitting an update packet to the server. Server gets data from every client, controls game progress, sends update packets to every client separately. There is no direct network communication among all clients in Quake.

Game traffic experiment was performed in a test-bed (a separated LAN), which only runs the test application and nothing else. All Win32 PC workstations in LAN test-bed are state of the art. In order to decrease CPU burden of game server host, Quake’s “dedicate” mode is used in experiment. All other selectable parameters are selected differently in different game sessions to increase the coverage of empirical data. Three game sessions are played in experiment, every session lasts a little more than 10 minutes. Games are played by four or five players (clients) in the experiment. All players are skillful, and only empirical data while all players were competing one another in battlefield situations are used for analysis, in order to ensure typical “game-in-process” traffic.

In order to obtain network traffic, a packet sniffer is essential. Packet sniffer is a program that records all of the traffic that the network card in a computer sees. Ethereal is a free sniffer and network protocol analyzer for both Unix and Windows. It examines data from a live network or from a capture file on disk. Ethereal’s time precision is 1 micro second and it is the highest time precision among all sniffers which we know. It is open source (many other sniffers are not) and users can develop programs to manipulate empirical traffic data. Dave Laporte et al [Claypool et al. 03] made comparison among several well-known sniffers and recommended Ethereal.

Every game client (or server) host in the test-bed is installed with sniffer, every Ethereal program sniffs only the outbound traffic of the same computer. In this way packets send time (and inter-send time) are accurately sniffed.

6. ANALYSIS METHODS

Analyzing steps are: Depict traffic histograms; Choose appropriate analytic distribution by eyeballing; Used curve fitting program to gain parameters of probability distribution; Measure discrepancy (goodness-of-fit test).

6.1. Obtain Fully Defined Analytic Functions

Before deciding empirical data fit to some specific statistic model, we must determine the analytic distribution for empirical data fitting. It is done with histograms by eyeballing.
Once a distribution is chosen, distribution’s parameters should be determined before we carry out goodness-of-fit test. Gnuplot is a data plotting program with curve fitting parameter determining function. Gnuplot can provide fully defined fitting function for empirical data.

6.2. Probability Distributions

The extreme and deterministic distributions are important in game traffic modelling in reference. The general formula for the probability density function of the extreme distribution is:

\[
f(x) = \frac{1}{\beta} \cdot e^{-\left(\frac{x - \mu}{\beta}\right)} \cdot e^{-\left(\frac{-x}{\beta}\right)}
\]

Deterministic distribution is the distribution determined by a constant value.

By eyeballing, server packet size is not only extreme distribution, but also Weibull and Lognormal distributions. We tried to see the effects of fitting the empirical data by Weibull and Lognormal distributions in Gnuplot and found Weibull and Lognormal distributions fitted the empirical data no better than the extreme distribution. Furthermore, Weibull and Lognormal distribution have three parameters, the shape parameter (extreme distribution does not have shape parameter) makes the PDF and CDF functions complicated. The change of PDF curve of each distribution is not continuous in some points of shape parameter value. The non-continuousness may add complexity to the fitting conclusions. Therefore, we only use the extreme distribution for the test and modelling.

6.3. Goodness Of Fit Test

Even if the discrepancy between analytic and empirical distributions is small by eyeballing, the fitness between both distributions is un-determined. Gnuplot’s result cannot be used for determining confidence levels because its deviation estimating is over-optimistic. What test should we use? Borella [Borella 00] considered traditional goodness-of-fit methods such as, Chi-square and Kolmogorov-Smirnov tests are not ideal for Internet traffic, he used other method instead. However, we intended to use the simplest test method and decided not to give up traditional methods until all of them fail.

We did use Chi-square test for all our server packet size empirical data, but every statistical “null hypothesis” is rejected no matter how we adjusted the bins of Chi-square test. It is because heavy upper tail yields very large Chi-square value by examining test details. How can we solve this problem?

One main limitation of Kolmogorov-Smirnov (K-S) test is that it tends to be more sensitive near the centre of the distribution than at the tails. This limitation might help to gain certain non-rejected hypothesis (otherwise our conclusion is “irregular traffic pattern”) in spite of the obstacle of heavy tail. However, it is by no means that K-S test is mathematically un-strict. The vertical difference of ECDF-CDF point-pair is checked at every x-axis point with unit-increment (i.e. 1 byte), to make sure this difference is satisfactory at every possible value (in our case, every value of packet size). In contrast, users are allowed and even suggested to adjust bins to gain more satisfactory results in chi-square test.

6.4. The Use Of K-S Test

Our empirical data of every single session/client usually contain thousands of packets. By definition, the “critical value” of K-S test is dynamic and equals to the mathematic expression “c/sqrt(n)”, where c is a constant (c depends on significance level, and is 1.22 on significance level 10%) and n is number of samples. The critical value decreases if n increases. The larger the number of samples is, the more likely K-S test leads to “rejected” result. It is almost impossible to achieve “non-rejected” results if we use whole data in one single K-S test, because the number of packets is large.

In order to solve this problem, we use relatively small number of samples (or sample-size) for K-S test, and all samples are randomly selected across the whole range of empirical data. First, we determine sample-size for the K-S test, for example, assuming sample-size is 100. Then, the test with the same sample-size (i.e. 100) is repeated 30 times (we name it 30-repetition test) with totally random selected samples. That is, one test uses 100 random samples, one 30-repetition test uses 3000 random samples. Thirdly, we choose ten different sample-sizes for 30-repetition test, and they are 36, 49, 64, 81, 100, 121, 144, 169, 196, and 225. Finally, every 30-repetition test yields 30 D-values, we would expect the number of rejects to be between 1 and 5 (inclusive) using 10% significance level. That is, we would expect the number of non-rejects to be no less than 25, in order to get non-rejected null hypothesis.

7. RESULTS

7.1. Server Packet Size

The curves of server packet size are shown in Figure 1, 2 and 3. The single fitting curve (a curve of the extreme distribution) in every figure is fits to only one empirical curve in the same figure.

Does server packet size fit to the extreme distribution? We cannot answer this question until K-S tests are performed. Server packet size empirical data are from 13 individual session/client. Data of every session/client undergo K-S test with 10 different sample-sizes (i.e. 36, 49, 64, 81, 100, 121, 144, 169, 196, 225). Every sample-size undergo 30-repetition test. The number of non-rejects in a single 30-repetition test is an integer between 0 and 30. The number of non-rejects between 25 and 30 illustrate we would not reject “null hypothesis”.

The results of the whole test process (of last paragraph) are shown in Table 1 (in fact, we have done the whole test process 10 times, and all results are similar). The number of non-rejects between 25 and 30 are marked with asterisk. Most items in column (sample-size) 36, 49, and 64 are marked with asterisk so they support “extreme distribution”, and there are many non-rejected items between sample-size 81 and 225. The unsatisfied items in low sample-size columns (column 36, 49, and 64) are caused by outliers, by examining the details of K-S test processes. The proportion of the number of packets of
outliers is small, comparing to the total number of packets. If we ignore the outliers, these a few un-satisfied items will disappear.

Figure 1: Server packet size in Session 1 (4 clients)

Figure 2: Server packet size in Session 2 (4 clients)

Figure 3: Server packet size in Session 3 (5 clients)

According to Table 1, under the condition that some outliers are ignored, the “null hypothesis” of the extreme distribution (for all server packet size) is more likely to be accepted with smaller sample-sizes (36, 49 and 64); If larger sample-sizes (225, 196, 169, 144, 121, 100, 81) are considered, only a portion of server packet size (for example, s1/c1, s1/c2, s2/c2) is the extreme distribution. We also tried sample-sizes less than 36, the results are much better than sample-size between 36 and 225.

Table 1: Server Packet Size K-S Test Results

<table>
<thead>
<tr>
<th></th>
<th>36</th>
<th>49</th>
<th>64</th>
<th>81</th>
<th>100</th>
<th>121</th>
<th>144</th>
<th>169</th>
<th>196</th>
<th>225</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1/c1</td>
<td>27*</td>
<td>28*</td>
<td>30*</td>
<td>27*</td>
<td>26*</td>
<td>25*</td>
<td>24</td>
<td>27*</td>
<td>28*</td>
<td>24</td>
</tr>
<tr>
<td>s1/c2</td>
<td>27*</td>
<td>28*</td>
<td>30*</td>
<td>27*</td>
<td>26*</td>
<td>25*</td>
<td>24</td>
<td>27*</td>
<td>28*</td>
<td>26*</td>
</tr>
<tr>
<td>s1/c3</td>
<td>26*</td>
<td>27*</td>
<td>22</td>
<td>23</td>
<td>26*</td>
<td>29*</td>
<td>27*</td>
<td>25*</td>
<td>22</td>
<td>20</td>
</tr>
<tr>
<td>s1/c4</td>
<td>30*</td>
<td>27*</td>
<td>26*</td>
<td>23</td>
<td>25*</td>
<td>21</td>
<td>22</td>
<td>21</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>s2/c1</td>
<td>28*</td>
<td>26*</td>
<td>27*</td>
<td>27*</td>
<td>26*</td>
<td>25*</td>
<td>23</td>
<td>24</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>s2/c2</td>
<td>26*</td>
<td>27*</td>
<td>26*</td>
<td>27*</td>
<td>28*</td>
<td>26*</td>
<td>26*</td>
<td>25*</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>s2/c3</td>
<td>27*</td>
<td>27*</td>
<td>27*</td>
<td>27*</td>
<td>26*</td>
<td>26*</td>
<td>25*</td>
<td>22</td>
<td>21</td>
<td>22</td>
</tr>
<tr>
<td>s2/c4</td>
<td>25*</td>
<td>27*</td>
<td>27*</td>
<td>27*</td>
<td>26*</td>
<td>25*</td>
<td>21</td>
<td>20</td>
<td>19</td>
<td>10</td>
</tr>
<tr>
<td>s3/c1</td>
<td>26*</td>
<td>25*</td>
<td>23</td>
<td>25*</td>
<td>26*</td>
<td>21</td>
<td>27*</td>
<td>21</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>s3/c2</td>
<td>26*</td>
<td>26*</td>
<td>26*</td>
<td>26*</td>
<td>26*</td>
<td>22</td>
<td>26*</td>
<td>21</td>
<td>28*</td>
<td>24</td>
</tr>
<tr>
<td>s3/c3</td>
<td>28*</td>
<td>27*</td>
<td>26*</td>
<td>19</td>
<td>24</td>
<td>21</td>
<td>19</td>
<td>20</td>
<td>17</td>
<td>16</td>
</tr>
<tr>
<td>s3/c4</td>
<td>26*</td>
<td>22</td>
<td>23</td>
<td>20</td>
<td>27*</td>
<td>26*</td>
<td>22</td>
<td>21</td>
<td>22</td>
<td>18</td>
</tr>
<tr>
<td>s3/c5</td>
<td>27*</td>
<td>28*</td>
<td>23</td>
<td>25*</td>
<td>28*</td>
<td>25*</td>
<td>25*</td>
<td>23</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

7.2. Client Packet Size

The distribution of client packet size (Figure 4) is difficult to model although the traffic itself is much simpler than server packet size. The range of client packet size is small (33 to 68), there are one high peak (43) and 10 low-equal-height peaks (39, 41, 45, 47, 49, 51, 53, 55, 57, 59), interval of every 2 sequential peaks is 2. A rough bi-model is:

- The first part is the high peak, it is a deterministic distribution f(x).
- The second part is the rest traffic g(x).

The client packet size is fit to a bi-model distribution h(x):

\[ h(x) = c \cdot f(x) + d \cdot g(x) \]  

where \( c, d \) are constants, \( c>0, d>0, c+ d = 1 \), \( f(x) \) is deterministic, \( g(x) \) is uniform distribution.

Figure 4: The distribution of client packet size

7.3. Inter-Send Time

Server inter-send time (Figure 5) is deterministic (two major peaks at 47.5 and 53 ms, plus a minor peak at 100ms in some cases). Client inter-send time (Figure 6) is deterministic distribution (at 33.1 ms), too. However, the height of peaks differs among different session/clients.

7.4. Conclusions

Quake’s inter-send time is a deterministic distribution; Client packet size is very simply distributed whereas it may not fit to any simple model; Server packet size is distributed
with many outliers, the main parts of empirical curves fit to extreme distribution in case outliers are ignored.

**Server traffic statistic values:** average inter-send time is 52.9ms, average packet rate is 18.9 packets per second, average bit rate is 11.1 KBPS.

**Client traffic statistic values:** average inter-send time is 37.1ms, average packet rate is 26.9 packets per second, average bit rate is 10.4 KBPS.

Game servers usually offer selectable options such as connection parameters, battlefield situations, etc. The different selections may lead to different results.

Game host CPU speed has direct influence upon traffic characteristics such as inter-send time and packet size.

We selected skillful game players, used only traffic during fierce battle situations. Some other researches used all traffic from and into a game server in a period time.

Some research did not measure the discrepancy between empirical and analytic data [Bangun et al. 00]. Some researches did not offer detail probability distribution modelling. [Feng et al. 02] [Joyce 00] [Claypool et al. 03].

Details in modelling determinant distribution are not given in reference. We have noticed two problems while fitting empirical data to determinant distribution: Height of peak differs among different empirical session/clients; There are traffic fall fully outside the range of the determinant distribution’s peak and such out-range traffic are non-ignorable in some cases. Therefore, our conclusions about determinant distribution are not strictly accurate.

### 8.3. Different Average Values

Empirical data of every individual session/client usually contain more than 10000 packets, we considered such amount of data should have yielded same average values among different session/clients. However, the average packet rate, bit rate and packet size differ slightly one session (and even client) from another. Reasons may be different game server configurations, and the skills of different players.

### 9. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the European Community through grants IST-2001-35354 and ESE/EST/ERDF/03/13/0033.

### REFERENCES


UNDERSTANDING RISK TAKING BEHAVIOR OF RECKLESS DRIVERS THROUGH FUZZY LOGIC

Tarik Ozkul
Computer Engineering Department
American University of Sharjah,
Sharjah, UAE
E-mail: tozkul@aucehsarjah.edu

KEYWORDS
RTA, MVA, Traffic accidents, Fuzzy modeling, Behavior analyses

ABSTRACT
A fuzzy logic model is generated for determining an index which reflects the speeding propensity of select group of drivers. Through the model we would like to understand the behavior pattern of drivers, what tempts them to speed and what urges them to slow down. The model will eventually be helpful for developing methods to curb moving vehicle accidents, (MVA) in UAE. The model generates an index, which is called speed propensity index, which reflects the propensity of driver to speed. The model has many fuzzy inputs that cause the index to increase and decrease. Unlike traditional methods that provide crude understanding of the behavior, this method generates continues index value which can be used for measuring effectiveness of countermeasures on the driver behavior.

INTRODUCTION
UAE have modern road network which is comparable to the best road networks in the world and equally well developed traffic sign system. Driver education and licensing procedure is meticulous and up to or above international standards. Yet the number of traffic accidents in UAE is much higher than that of countries with comparable road networks. Death rate per 100,000 vehicles in UAE is more than 6 times higher than that of UK and USA. The road traffic accidents (RTA) studies indicated that 90% of the accidents attributed to road user characteristics. What is more alarming is that 80% of these accidents are caused by drivers in the age group of 18-32 and mostly related to speeding. MVA is the seconds leading cause of death in UAE second to coronary disease, and leading cause of death in the age group of 18-32 (Sankaran 1997).

The problem has gotten so noticeable that at American University of Sharjah, which has a relatively small student body of 3500, we loose about 10 students a year to traffic accidents. UAE authorities have tried many different techniques to reduce the traffic accidents yet the accident rate keeps climbing (Gulf news 2003).

In this study we wanted to examine the behavior of group of youth who are likely to have speeding accidents to see what makes them susceptible to traffic accidents and how it can be prevented. We wanted to construct a detailed model of behavior pattern based on fuzzy logic. Although constructing a fuzzy model seems like overkill for this problem, we wanted to see and compare which measures are more effective in controlling the behavior of drivers under study.

The input variables to the model are variables which are fuzzy in nature. The model that needs to be constructed should be sensitive enough to show the effects of different countermeasures used by traffic police. For this reason we decided to develop a fuzzy model for a specific group of youth which we think that they are likely to get involved in speeding accidents. The model generates a speed propensity index which indicates the speeding behavior of the person. The model is constructed using membership functions and generates an index value under different conditions.

METHOD
To construct the fuzzy logic model we needed membership functions of different parameters indicating their effect on speeding behavior of the subjects. Fuzzy if-then rules are also needed to simulate the decision process of these individuals. Since the study involved collecting detailed membership functions from subjects regarding many fuzzy input variables, the surveys conducted were considerably different than ordinary questionnaires. The respondents needed to be educated on the theory of fuzzy logic, membership functions and how to generate the membership functions carefully.

The students who are registered in COE 494 Soft computing class of spring 2004 term were educated on the subject of fuzzy logic and membership functions and each student is asked to collect data for five different respondents. The students were not told the purpose behind the study, and how the data is going to be used. They were given the assignment as a study of generating membership functions.

Each questionnaire had eight membership functions to be generated by the subject. The respondents were asked to draw membership functions regardless how,

- the quality of the car,
- the quality of the road,
- the traffic on the road,
- the number of passengers in the car,
- the effect of tailgating drivers,
- the existence of police,
- the existence of speed radars,
- the effect of seeing accidents on the road

effected their speeding behavior.
The data collection effort has resulted in 114 responses which are mostly from the age group of 18-32. The responses are analyzed carefully and few responses among them which have contradictory and unrealistic data are eliminated. Majority of the subjects was found to have acceptable driving habits with understandable behavior. Out of the 114 responses, nine of them have appeared as the category which is likely to get involved in accidents. What makes this category different was;

a) Their tendency to over speed much more that the rest of the respondents,

b) Their tendency to get involved in racing behavior whenever they are tailgated by other drivers.

The common characteristic of this group of drivers is their tendency to speed further as more drivers involved in the racing action. It is decided that this is the category of youth likely to get involved in speeding accidents so we based our model on the behavior of this group. Their responses are averaged in order to get a better representation of behavior.

**FUZZY LOGIC APPROACH**

Fuzzy logic is chosen as the tool for development of model. Fuzzy logic, which is also called “fuzzy set analyses” and “possibility theory” can deal with uncertainty and imprecision. Fuzzy logic is a very effective tool to work with problems which has no clear boundary and definitions. Fuzzy logic should not be confused with probability theory which deals “likelihood of events”. Unlike Boolean set which defines either inside or outside a set, fuzzy logic deals with membership functions which can be partially inside or outside a set. Figure 1 indicates this concept.

![Figure 1: Comparison of Boolean and Fuzzy sets.](image1)

In a fuzzy set the position of the element within the set is described by a membership function (\(\mu\)). \(\mu=1\) indicates that the element is completely in the set. \(\mu=0\) indicates that the element is completely outside the set. Any value of \(\mu\) between 0 and 1 indicates that the element is partially contained in the set. (Yen and Langari, 1999)

Fuzzy logic modeling is used successfully in many applications where complex decision is needed to be made (Gentile et al 2003). The process mimics decision process of an expert person making decision. In this application fuzzy logic modeling is used to mimic the behavior of driver. The output index generated by the model indicates the speeding behavior of the driver and ranges from 0 to 1. In our model, the score of 0.5 indicates the legal speed limit in a given situation. In a 120 km/hr legal speed zone, the speed index of 0.6 corresponds to 140 km/hr, whereas index of 0.9 corresponds to speed of 210 km/hr.

**LINGUISTIC VARIABLES OF THE MODEL**

In fuzzy logic the variables are linguistic variables which are divided into sub-ranges, where these sub ranges overlap with the neighboring sub-ranges. The outputs of fuzzy systems can also be linguistic variables. Input output relationships are defined by if-then rules. Depending on the output generation fuzzy inference system can be either Mamdani type or Sugeno type (Jang et al 1997). The model has been constructed using eight fuzzy variables and Sugeno model. Matlab fuzzy logic toolbox is used for implementation of the model.

The membership function for quality of the car is given in Figure 2. The cars are divided in to three categories which are labeled as compact, midsize and sporty.

![Figure 2: Quality of the Car Membership Function](image2)

The membership function for quality of the road is given in Figure 3. The road condition is expressed in three categories which are labeled as poor, OK and good.

![Figure 3: Quality of Road Membership Function](image3)
The membership function for traffic on the road is given in Figure 4. The traffic condition is expressed in three categories which are labeled as heavy, OK and light.

Figure 4: Quality of the Road Membership Function

The membership function for passengers in the car is given in Figure 5. The passenger condition is expressed in three categories which are labeled as few passengers, one passenger and only driver.

Figure 5: Number of Passengers Membership Function

The effect of tailgating drivers is investigated through membership function given in Figure 6. The categories are, none, one and few.

Figure 6: Number of Passengers Membership Function

The driver reaction to possibility of police patrols on the road is investigated through membership function given in Figure 7. The categories are; none, possible police patrol and definite police patrol.

Figure 7: Police Patrol Membership Function

The driver reaction to possibility of radar speed detectors on the road is investigated through membership function given in Figure 8. The categories are; none, possible and definite availability of radar speed traps.

Figure 8: Radar Speed Trap Membership Function

The driver reaction to seeing accidents on the road is measured through membership function given in Figure 9. The categories are; none, one accident and few accidents on the road.

Figure 9: Accidents on the road Membership Function

Figure 10 shows the overall list of inputs and outputs of the Fuzzy Toolbox package.

FUZZY IF-THEN RULES

The model is based on Sugeno approach with total of 24 if-then rules. The Figure 11 below shows some of the rules as they are implemented in the Matlab Fuzzy Toolbox package.
Output of the model and the 3-D output curve are given in Figures 12 and 13 respectively.

**CONCLUSION**

The membership functions and the model has provided important information about the behavior of the youngsters. Highlights of this behavior are as follows:

The drivers tend to speed up when driving sporty cars and driving on good roads. The drivers also tend to slow down when there are additional passengers in the car. The drivers in this risk taking category tend to speed up when they are followed by other drivers. The probability of existence of police patrols on the road slows down the drivers. The existence of speed radar traps also slows down the drivers. The police patrols are found to be more effective in discouraging drivers from speeding than the radar speed traps. The accidents seen on the road brings the drivers into their senses resulting in driving at or below legal speed limits.

Considering that 0.5 index value corresponds to legal speed limit, typical speeding behavior in this category is given in Table 1. (The speed value is generated for 120 km/hr legal speed limit zone for ease of comparison.)

Studying the data given in Table 1, it may be possible to generate countermeasures other than speed radars and police patrols. It seems that drivers tend to slow down upon seeing accidents on the road which reminds them negative consequences of speeding behavior. Since this seems like a very effective method, it may be advisable to put billboards on the roadside to warn drivers about deadly consequences. Another approach may be to put cars involved in accidents on permanent display near the road side with adequate warning messages.

We hope that this study provides some help in curbing accidents resulting senseless deaths among youngsters.
Table 1: Conditions of Driving Versus Speed for Risk Taking Group of Drivers

<table>
<thead>
<tr>
<th>Conditions of driving</th>
<th>index</th>
<th>speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driving compact car</td>
<td>0.54</td>
<td>130</td>
</tr>
<tr>
<td>Driving midsize car</td>
<td>0.61</td>
<td>144</td>
</tr>
<tr>
<td>Driving sporty car</td>
<td>0.75</td>
<td>180</td>
</tr>
<tr>
<td>Poor road condition</td>
<td>0.3</td>
<td>72</td>
</tr>
<tr>
<td>OK road condition</td>
<td>0.53</td>
<td>127.2</td>
</tr>
<tr>
<td>Good road condition</td>
<td>0.76</td>
<td>182.4</td>
</tr>
<tr>
<td>Heavy traffic condition</td>
<td>0.3</td>
<td>72</td>
</tr>
<tr>
<td>Normal traffic condition</td>
<td>0.5</td>
<td>120</td>
</tr>
<tr>
<td>Light traffic condition</td>
<td>0.81</td>
<td>194.4</td>
</tr>
<tr>
<td>With full passengers</td>
<td>0.48</td>
<td>115.2</td>
</tr>
<tr>
<td>One passenger</td>
<td>0.68</td>
<td>163.2</td>
</tr>
<tr>
<td>Driver alone</td>
<td>0.78</td>
<td>187.2</td>
</tr>
<tr>
<td>No tailgating</td>
<td>0.51</td>
<td>122.4</td>
</tr>
<tr>
<td>One car tailgating</td>
<td>0.68</td>
<td>163.2</td>
</tr>
<tr>
<td>Many cars tailgating</td>
<td>0.87</td>
<td>208.8</td>
</tr>
<tr>
<td>No police patrol</td>
<td>0.75</td>
<td>180</td>
</tr>
<tr>
<td>Police patrol possible</td>
<td>0.56</td>
<td>134.4</td>
</tr>
<tr>
<td>Definite police patrol existence</td>
<td>0.39</td>
<td>93.6</td>
</tr>
<tr>
<td>No speed radar</td>
<td>0.81</td>
<td>194.4</td>
</tr>
<tr>
<td>Speed radar possible</td>
<td>0.61</td>
<td>146.4</td>
</tr>
<tr>
<td>Definite speed radar possibility</td>
<td>0.42</td>
<td>100.8</td>
</tr>
<tr>
<td>No accidents seen on the road</td>
<td>0.763</td>
<td>183.12</td>
</tr>
<tr>
<td>One accident seen on the road</td>
<td>0.563</td>
<td>135.12</td>
</tr>
<tr>
<td>Multiple accidents seen on the road</td>
<td>0.4</td>
<td>96</td>
</tr>
</tbody>
</table>

REFERENCES


TARIK OZKUL has received his BSc in Electrical Engineering from Bogazici University, Istanbul/Turkey in 1981 than received his MS and PhD in Electrical and Computer Engineering from Florida Institute of Technology in years 1984 and 1989. After being in industry for sometime, he has joined Computer Engineering Department of American University of Sharjah, Sharjah/UAE in year 2001 where he has been teaching since then.
ON THE NEED FOR HUMAN PERFORMANCE MODELLING IN THE SIMULATION OF A BALANCED PRODUCTION LINE

Peter Lawrence
School of Business Systems
Monash University, Clayton Campus
Victoria 3800 AUSTRALIA
Email: peter.lawrence@infotech.monash.edu.au

ABSTRACT

It has been observed that stochastic discrete event simulations of systems involving human workers demonstrate considerably less accuracy than simulations involving deterministic automated work cells and that this highlights the inadequacy of current methods of modelling human performance (Baines and Kay 2002). This study investigates the impact of two proposed sources of human performance variability on production volume of a balanced flow line, time of day effects and individual worker efficiency, and assesses the accuracy of a traditional, time- and worker-invariant simulation modelling approach on such a system. It was found that, in the system studied here, ignoring the variant components of work cell cycle times produces modelling inaccuracy in some cases but that these errors lie within limits that many modellers would consider acceptable.

A possible explanation for this discrepancy, which Baines and Kay explore, is that human behaviour, in particular work performance, is neither stationary nor consistent between individuals. In terms of DES methodology this implies that, rather than the traditional stochastic modelling approach of a single time-invariant distribution for each work cell, the model should incorporate a different, time-dependent distribution for each individual worker for each work cell.

A considerable amount of effort has gone into developing an understanding of, and even models for, human performance variation, particularly in the Psychology literature. However, though Baines et al. (2003a, 2003b) cite some useful work in the area, a reliable model of human performance suitable for DES work cell applications appears still to be some time away.

Given the difficulty involved in modelling these variations in human performance this study aims to look at how important some of these variations are in affecting the accuracy, and hence utility, of a DES model of a balanced production line.

INTRODUCTION

For close to half a century Discrete Event Simulation (DES) has been used to model factory production lines and improve manufacturing operations management (Kelton, Sadowski and Sadowski 1998:13, Robinson 1994:3). While the accuracy and reproducibility of fully automated work cells allows simulation models to be deterministic the involvement of human workers creates complications. However, the flexibility afforded by human workers assures their presence in the factory for some time into the future and so addressing their impact on factory operations is vital.

The primary effect of human involvement in DES is the need to model the irregular timing of human activity such as arrival times and work cell cycle times. This is generally achieved by replacing the fixed timing characteristic of automated systems with a stream of pseudo-random numbers generated by an appropriate statistical distribution and thus developing a stochastic rather than deterministic DES model.

However, while this is the standard approach it has been observed by Baines and Kay (2002) that while DES results for automated systems are generally good they are very often inaccurate for systems involving human workers.

The accuracy and reproducibility of fully automated work cells allows simulation models to be deterministic the involvement of human workers creates complications. However, the flexibility afforded by human workers assures their presence in the factory for some time into the future and so addressing their impact on factory operations is vital.

HUMAN PERFORMANCE VARIATION

There are many ways in which the behaviour, and hence performance, of human workers vary from that of machines but this study concentrates on two particular characteristics that are both well accepted and have a direct impact on the time taken to perform a task. They are time of day effects and individual worker efficiency (Smith 1989, 1992).

Time of Day Effects

It is known that many human biological functions and characteristics, such as body temperature, follow a natural cycle and it is generally accepted that this impacts behaviour and performance in some manner. Work in this area appears to have begun early in the twentieth century and was aimed at establishing the optimum time of day for academic teaching (Carrier & Monk 2000).

Aschoff & Wever (1962: cited in Scott 1994) established that these biological rhythms have a period of about 25 hours but they are entrained to the 24-hour day by external time cues. This natural chronobiological cycle has become commonly known as the circadian rhythm (Scott 1994).
Scott cites much evidence to suggest a connection between this circadian rhythm for body temperature and human performance. Tasks with a small cognitive load seem to have a performance variation that tracks the circadian body temperature cycle (Kleitman 1963) while high memory load tasks are phase-shifted from this cycle. Folkard et al. (1976) suggested high memory tasks to be in anti-phase with the temperature rhythm and Monk’s (1982) analysis used Folkard’s data to postulate a medium to high memory load performance cycle that peaks 3-hours ahead of the low load performance cycle.

Monk also noted that superposed on this circadian effect is the additional factor of number of hours worked. Smith (1989, 1992) also discusses the impact of meals on performance including the so called “post-lunch dip”.

All of this suggests that, if human performance measures such as cycle times can be adequately modelled by pseudo-random numbers from some statistical distribution the parameters of that distribution are unlikely to be time-invariant.

**Individual Worker Efficiency**

Clearly not all workers are identical and so it is not expected that they will perform with equal efficiency. Baines et al. (2003a, 2003b) postulate that this will have an impact on model performance.

To improve line balance (and worker attention and motivation) it is common for workers to regularly rotate work cells. Thus, empirical data gathered over several days at a particular work cell, to be used in fitting appropriate statistical distributions for the DES model, will aggregate across numerous workers. While this is not ideal it is generally not possible to model individual workers, not just because of the extra effort required but often also for industrial relations reasons. The Baines et al. (2003) study installed an automated system that gathered work cell cycles times over a prolonged period on a major automotive production line but it was not possible to identify individual worker performance for this reason. Thus the distributions used in a stochastic DES model will usually be integrated over the entire workforce employed at the time data gathering occurred.

**Other Factors**

There are a number of other factors that differentiate the performance of human workers from machine. Behaviour such as working ahead, for example, is not uncommon. This is where a worker, having completed a task but unable to move the work piece on due to line congestion downstream, may physically move upstream and start working on a work piece that the DES model would say has not yet arrived at the work cell for processing.

However, the aim of this study is to concentrate on just two factors in human performance circadian time-variance and individual worker efficiency.

**METHODOLOGY**

Two models were created of an identical balanced factory production line. The only difference between the two models was that the first, referred to as the $\Psi$ model had work cell cycle times that included a circadian component of amplitude $\gamma$ and a variation in worker efficiency $\eta$. Variations on this model are thus referenced to as $\Psi_{\gamma,\eta}$.

The second model, referred to as the M model, had work cell cycle times that were best-fit, time-invariant statistical distributions based on empirical data taken from 5-8-hour shifts of the corresponding $\Psi$ model. Thus $\mathcal{M}_{\gamma,\eta}$ is the model with time- and worker-invariant cycle time distributions based empirically on the 5-shift performance of $\Psi_{\gamma,\eta}$.

To assess the M model’s success at predicting the performance of the $\Psi$ model each was run for 1000×8-hour shifts and confidence interval estimates made of the difference in production volume based on the null hypothesis that there is no difference between the production capacity of the M and $\Psi$ models.

All models were built using the Arena simulation package and empirical distribution fittings was performed using Arena’s inbuilt Input Analyser.

**The Factory**

The factory modelled in this study was a hypothetical balanced, linear flow line consisting of 9 discrete work cells which each work piece visits in turn. The layout is shown in figure 1. The first work cell (top left) takes raw materials as required (assumed to be always available) and process the work piece with the aid of a single human worker who must be present for the entire cycle time. When processing is complete the work piece travels down an accumulating conveyor to the next cell where it is similarly processed by a different worker, and so on until all nine work cells have been visited at which point the finished product leaves the system (bottom left). The characteristics of all work cells are the same, but necessarily all workers.

The work cells are spaced along the accumulating conveyor such that there is only room between cells for a single work piece to wait. Thus, if any worker falls more than one unit behind the pace then upstream cells will be unable to commence processing their next work piece.

The factory operates on a single 8-hour shift each day and, for simplicity of modelling, there are no breaks during the shift. Each worker is assigned to a particular work cell at the commencement of the shift but, to improve balance, workers rotate work cells every hour.

**Modelling Human Performance**

The objective of this study is to determine the impact of some typical human performance variability on the
accuracy of simulation modelling and so the aim here is not to produce an accurate model of human performance in any given situation but one that incorporates plausible variations in work cell cycle time as a function of time of day and individual worker efficiency.

Cycle times were pseudo-random variates taken from a lognormal distribution with a mean ($\mu$) of 0.9 mins. and a standard deviation of 7% of the mean (0.063 mins.) This gives a 3σ range of ±21% which accords with the 20% variability target from the company work-study convention reported by Baines et al. (2003).

The time of day component was modelled by varying the mean of the lognormal distribution over the course of the shift as a sinusoid, as is commonly the case with modelling circadian rhythms (Scott 1994), with an amplitude $\gamma$ about a mean of 0.9. More specifically,

$$\mu = 0.9 \left[ 1 + \gamma \cos \left( \frac{\pi t}{1440} \right) \right]$$

where $t$ is time measured in minutes.

To model worker efficiency the mean of the lognormal distribution was further modified by multiplication by the efficiency coefficient ($\eta$) of the individual worker involved. Thus,

$$\mu = 0.9 \left[ 1 + \gamma \cos \left( \frac{\pi t}{1440} \right) \right] \eta$$

Three further model pairs were then built and tested as follows:

- $M_{0.1,1}/\Psi_{0.1,1}$ to determine the effect of circadian variation. In this case worker efficiency remained invariant at unity but the circadian cycle imposed a variation with 10% amplitude on the mean cycle time of each work cell. This meant that, over the course of the 8-hour shift, the mean cycle time varied from 0.99 down to 0.855 mins.

- $M_{0.1,0}/\Psi_{0.1,0.1}$ to determine the effect of variable human work efficiency. In his case there was no circadian variation but the 9 workers were assigned different values of $\eta$, symmetrically distributed about unity with a maximum variation of 0.1. Specifically, the chosen values were {0.9, 0.93, 0.94, 0.98, 1.0, 1.02, 1.06, 1.07, 1.1}.

- $M_{0.1,1,0.1}/\Psi_{0.1,1,0.1}$ to determine the combined effect of both the circadian and worker efficiency variations described above.

**RESULTS**

Table 1 and figure 2 show the resultant production figures for each of the models expressed as a 99% $t$-statistic confidence interval. The use of 1000 replications to determine the mean production volume justifies, via the Central Limit Theorem, the use of the $t$- (or $z$-) statistic and also explains the small size of the intervals.

<table>
<thead>
<tr>
<th>Model</th>
<th>N</th>
<th>1/2 width</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Psi_{0.1}$</td>
<td>448</td>
<td>0.0732</td>
</tr>
<tr>
<td>$\Psi_{0.1,1}$</td>
<td>434</td>
<td>0.0703</td>
</tr>
<tr>
<td>$\Psi_{0.1,0.1}$</td>
<td>426</td>
<td>0.103</td>
</tr>
<tr>
<td>$\Psi_{0.1,0.1}$</td>
<td>412</td>
<td>0.0931</td>
</tr>
<tr>
<td>$M_{0.1}$</td>
<td>449</td>
<td>0.0708</td>
</tr>
<tr>
<td>$M_{0.1,1}$</td>
<td>433</td>
<td>0.0769</td>
</tr>
<tr>
<td>$M_{0.1,1,0.1}$</td>
<td>440</td>
<td>0.0971</td>
</tr>
<tr>
<td>$M_{0.1,1,0.1}$</td>
<td>426</td>
<td>0.0871</td>
</tr>
</tbody>
</table>

**Effect of Human Variation on System Performance**

Considering just the $\Psi$ models, it looks at first view that the introduction of both circadian rhythms and individual efficiency variations produce a marked drop in production within the system. However, at least in the case of circadian variation, this must be treated with some caution. Table 2 shows the mean (and standard deviation) for cycle times from each of the $\Psi$ models aggregated across all replications and work cells (approximately 4 million data points in each case).
The combination of these two variations results in an absolute drop in production volume of around 8% but, when this is normalised with respect to the mean cycle time this drops to around 5% making it comparable with worker efficiency variation alone.

It is worth noting that the mean cycle time for $\Psi_{0.1, 1}$ of 0.935 differs slightly from what might be expected from,

$$\mu(t) = 0.9 \left[ 1 + \gamma \cos \left( \frac{2\pi t}{1440} \right) \right]$$

$$\therefore \bar{\mu} = \frac{1}{\tau} \int_0^\tau 0.9 \left[ 1 + \gamma \cos \left( \frac{2\pi t}{1440} \right) \right] dt$$

where $\tau$ is the shift length (480 minutes).

$$\therefore \bar{\mu} = \frac{1440 \times 0.9}{2\pi \tau} \left[ u + \gamma \sin(u) \right]_0^{480}$$

$$= 0.9 \left[ 1 + \frac{1440}{2\pi \tau} \gamma \sin \left( \frac{2\pi \tau}{1440} \right) \right]$$

$$= 0.9372$$

As small as this discrepancy is it may seem at first surprising given the size of the sample that was averaged. However, the traditional mean calculation above is, in fact a biased determination of the mean in this case that will tend to overestimate. This is because it gives all values along the sinusoid equal weight when this does not occur in the simulation. The value of cycle time also determines the sampling rate during the simulation run and so fewer samples will be drawn when $\mu$ has a higher value. This has the effect of reducing $\mu$ below the expected value.

**The accuracy of the M Models**

Table and Figure 3 show the pairwise t-statistic 99% confidence interval estimates of the difference in production volumes between each M model and its corresponding $\Psi$ model, thus testing the Null Hypothesis that there is no difference between their production capacities.

**Table 3: Production Variation between M and $\Psi$ Models including 99% Confidence Interval Half-Width**

<table>
<thead>
<tr>
<th>Model</th>
<th>$\Delta N$</th>
<th>$\frac{1}{2}$ width</th>
<th>$\Delta N/N_{\Psi}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_{0.1, 1} - \Psi_{0.1, 1}$</td>
<td>0.144</td>
<td>0.0745</td>
<td>0.03%</td>
</tr>
<tr>
<td>$M_{0.1, 1} - \Psi_{0.1, 1}$</td>
<td>-1.26</td>
<td>0.0993</td>
<td>-0.29%</td>
</tr>
<tr>
<td>$M_{0.1, 1} - \Psi_{0.1, 1}$</td>
<td>14.2</td>
<td>0.141</td>
<td>3.3%</td>
</tr>
<tr>
<td>$M_{0.1, 1} - \Psi_{0.1, 1}$</td>
<td>14.3</td>
<td>0.125</td>
<td>3.5%</td>
</tr>
</tbody>
</table>
While the results for $M_{0,1} \sim \Psi_{0,1}$ reject the Null Hypothesis this does not imply the methodology is faulty. The empirical data generated from the 5 shifts of the $\Psi$ model naturally result in some difference between the $\Psi$ and M cycle time distributions. Since 1000 replications of a well balanced line such as this results in a very sensitive analysis (as evidenced by the very narrow confidence intervals) it is not surprising that this difference is detected. However, the fact that the difference is of the order of 0.03% suggests that $M_{0,1}$ does an excellent job of simulating the behaviour of $\Psi_{0,1}$.

Similarly, the M model accurately reflects the production volume of the corresponding $\Psi$ model for circadian variations with a small error of around 0.3%.

The discrepancy becomes greater as individual worker efficiency is included in the system but it is still quite small at around 3-4%. This magnitude of error is not insignificant but experience suggests that there are many practical modelling situations where it would be perfectly acceptable.

CONCLUSIONS

This study has suggested that the effect of system wide temporal variation in cycle time has little effect on the production volume of a balanced flow and that traditional DES methods can successfully model this with good accuracy. Non-uniform variations such as worker efficiency however, seem to have a larger effect on system performance and result in greater errors using traditional DES methodology. However, these errors were quite small and it may be true that a balanced line like this, which is typical of systems such as an automotive production line, can be modelled adequately without the inclusion of the human variations investigated here.

Nonetheless, this work should be viewed only as a pilot study and it is certainly not reasonable to generalise this conclusion to other systems without considerably more work. More needs to be understood about the sensitivity of these errors to factors such as system size, complexity and the magnitude and stability of human variation.

REFERENCES


A SYSTEM DYNAMICS APPROACH: HIGHER EDUCATION SYSTEM IN TURKEY

Zehra Kamışlı Öztürk
Anadolu University
Distance Education Faculty
Dekanlık, No:312
26470, Eskişehir,
Turkey
E-mail: zkamisli@anadolu.edu.tr

Servet Häsüll
Osmangazi University
Faculty of Engineering and Architecture
Department of Industrial Engineering
26030, Eskişehir,
Turkey
E-mail: shasul@ogu.edu.tr

KEYWORDS
Simulation, system dynamics

ABSTRACT
The purpose of this study is to develop a system dynamics model for the higher education system in Turkey (HES). With this model, we aim to attract attention to dynamic variables because of the fact that in the long term, this dynamics affects the number of students that will enter university. With system dynamics, the dynamic behavior over time of the HES has been better understood. In addition, we want to determine the fact that, feedback loops provide a better understanding about uncontrollable dynamics of the system.

INTRODUCTION
One key to a successful estimation is an effective system analysis. It helps to determine all the variables affecting the system. In the Turkish higher education system (HES), one of the tasks faced by managers is to estimate the number of students that will set in the universities. Because of the dynamics of the system, the numbers change from year to year. The complexity of the education system is increasing due to the impact of changes in economic factors, demography, and etc. Rapid changes in socio-economic structure of the world cause complex problems.

In this paper, a dynamic simulation model of HES is introduced. Simulation is used in a broad range of fields ranging from pure mathematics and the physical sciences, though engineering and computer science, business and the military, to economics and social sciences (Robinson 2002). We use system dynamics modeling which is one of the approaches to analyzing and modeling a complex system.

System dynamics dates back to the late 1950s, and interest in the methodology grew rapidly during 1960s and early 1970s. It is both a system enquiry methodology and a modeling approach (Polat and Bozdağ, 2001). System dynamics combines theory, methods, and philosophy for the understanding of management. It shows how things change over time (Forrester, 1998). Traditionally, system dynamics have relied very extensively on the use of intuition and experience by system owners and analysts to help design policies for improving system behavior over time (Keloharju and Wolstenhome 1989).

System dynamics is well suited to the analysis of problems whose behavior is governed by feedback relationships and that have a long-term time horizon. Foresters study on Urban Dynamics indicates that, system dynamics isn’t just a technique to provide solutions to specific problems that have been formulated, but it acts as a system methodology which determines the whole style of problem definition, qualitative analysis and change analysis.

In system dynamics methodology, the feedback structure of a system is described using causal loops (Smith and Ackere 2002). In system dynamics modeling, causal loop diagrams represent the major feedback mechanisms, which reinforce or counteract a given change in a system variable (Saysel and Barlas 2001). Reinforcing loops capture positive feedbacks while counteracting loops captures negative feedbacks.

THE HES SYSTEM AND ITS PROBLEMS’ OUTLINES
In Turkey, students enter universities by competitive entrance examination which is prepared and administered every year by the Student Selection and Placement Center (ÖSYM in Turkish).

There is a quota in formal education programs but there isn’t in distance education programs. Because of the supply and demand imbalance, it isn’t possible for every student enter to universities. Over quota is the main problem faced in this system. The foregoing statistics clearly show that, while the population of the country increased, the number of students increased (Gürüz 2003). Population isn’t only variable affecting the system, but also per capita GNP affects the system. In addition to GNP, economic factors are also in consideration because of the fact that GNP increases or decreases after economic crisis. Beside these dynamics there’re also some managerial dynamics like opening new departments, quotas’ changes in and etc.
Figure 1 shows the number of students that apply to the examination and the number of students assigned to the universities among the years 1997 and 2001. We call the number of students applying to the examinations as age population. Age population consists of grads from high schools, students of any university, and students of high school in the last class.

![Bar chart showing number of available places and demands between 1990–2001](image)

Figures 1: Number of available places and demands between 1990–2001

**SYSTEM DYNAMICS MODEL**

In order to recover from conventional decision mistakes, strategic planners and decision makers should consider feedback affects of the decisions. System dynamics methodology has been used to emphasize these effects. Because of the existence of numerous variables and interactions to handle, system dynamics seem to be the most proper approach for modeling in this study.

A system dynamics model structure consists of two main variables: stock and flows. Stocks are the state of a system and the flow variables are the control variables that regulate the flow into and out of the stocks.

For instance, in this paper one of the stock variables formal represents the number of students set to the formal universities.

The other variables used in our model are given here:

- **Age population** is the stock variable that represents the expected number of students that will take the exams every year.

- **Potential demand** is the stock variable that represents the number of students that refer to the examination.

- **Assigned students** are the stock variables that represent number students that set to the universities.

- **AÖF** is the stock variable that represents the number of students set to the distance education universities.

- **Acquisition ratio** is the ratio of number of students to number of potential demand.

- **Registration** is the flow variable that calculates the number of students that register to the universities.

Despite those variables the other ones are called converters.

Based on these variables, the next step is to build the causal-loop diagram and determine the feedback structure of the system. The dynamic feedback system is an ongoing set of relationships in which the output of an action taken by one person or group. The dynamic feedback systems approach to understanding management problems has been studied and extensively developed by Forrester (Roberts 1978).

Figure 2 shows the dynamic relationships between the variables. Age population increases as population increases. When the number of potential demand increases, the number of students acquires the exam increases. But when the number of registered students increases, age population decreases.

![Causal loop diagram of the HES](image)

Figures 2: Causal loop diagram of the HES

After constructing the feedback structure, all the things described below are represented as stock and flow diagram.

The stock and flow diagram of the HES is shown in Figure 3. The model is developed by using STELLA 6.0 software. The software supports the simulation, visualization, and analysis of system-dynamics models on PCs.

The formulations in the model are also constructed by the software. Initial values of the stock variables and the formulations of the converters are constructed by the users. The relationships are modeled mathematically in this way. The availability of visual modeling and simulation software also contributed significantly in making the methodology.

Some formulations used in this model can be given like this:
**VALIDATION AND ANALYSIS OF THE MODEL**

In system dynamics models, the objective of behavior is to check the ability of the model to generate behavior patterns similar to actual system behavior. Since system dynamics models seek to explain how and why the problematic dynamics are created, it is crucial that their structure be a valid representation of the real processes that play significant roles in creating the dynamics of concern (Saysel and Barlas 2001).

The model is simulated between the years 1990 to 2009. After the simulation run, the dynamic behaviors of the model have been obtained. Since it can be seen from the Figures 3–6, the numbers obtained from the model show great correspondence to the real numbers of the system. Thus, we can say that the model provides the behavior validation.

While building the model, some assumptions are used to simplify the modeling process. For instance, potential demand is coming from age population, and age population has three different inflows. But in the modeling process, we didn’t use them due to lack of sufficient data.

---

**Figures 3:** Stock and flow diagram of the model

**Figures 4:** Comparison between number of real demands and the numbers generated by the model

**Figures 5:** Comparison between number of real assigned students and the numbers generated by the model
CONCLUSION

The purpose of this study is to develop a system dynamics model for the higher education system in Turkey (HES). With this model, we aim to attract attention to dynamic variables because of the fact that in the long term, these dynamics affect the number of students that will enter universities. With system dynamics, the dynamic behavior over time of the HES has been better understood. In addition, we want to determine the fact that, feedback loops provide a better understanding about uncontrollable dynamics of the system.

With the dynamic models it’s possible to simulate and understand the consequences of different improvement alternatives (Olosson, Caldenfors, and Sixtensson 1997). More generally, in future works the model can be used as an experimental simulation laboratory. Thus, different scenarios, decisions or questions can be analyzed by a system dynamics based decision support system.

REFERENCES


Forrester, J., W. 1998. “Designing the Future” at Universidad de Sevilla, (December), Sevilla, Spain


BIOGRAPHY

ZEHRA KAMİŞLİ ÖZTÜRK was born in Eskişehir, Turkey and graduated from Osmangazi University Industrial Engineering Department in 2001. She has been working on her master thesis about simulation. She is interested in system design, simulation, and decision support systems. And she has been working in Distance Education Faculty of Anadolu University since 2002 as a research assistant.

SERVET HASIHL is an assistant professor of Industrial Engineering at Osmangazi University. He received his Ph.D. in Industrial Engineering from Osmangazi University. His research interests include simulation methodology, system design and manufacturing management. His recent work involves problems related to the deterministic and stochastic simulation-based optimization and nonlinear programming, with applications in logistics and production.
TEXTILE SIMULATION
PRODUCTION PLANNING AND SCHEDULING IN TEXTILE PROCESSES
SIMULATION IN THE TEXTILE INDUSTRY: PRODUCTION PLANNING OPTIMIZATION

a.k.a. PipPO

Gianluigi Ferraris*
Matteo Morini†

April 2004

Keywords:
Production, Scheduling, Optimization, Industrial Processes, Manufacturing

Abstract

PIPPO (Production Intelligence: Performing Planning Optimization) is an open project with the aim of supporting the crucial activity of deciding what is to be done, and when, within an industrial, applied, real-world situation. In other words: matching each and every task to an appropriate production unit, and deciding the priority every job is to be given. The problem, common to many different industries, arises when a considerable amount of different articles must be done on a relatively small number of reconfigurable units; typically topics like these, in the textile world, may have a strong impact on a fundamental issue: satisfying the always-in-a-rush customers, while keeping under control accessory production costs (set-up costs, machinery cleaning costs, ...) avoiding the dramatic losses related to a wasteful resources management, due to “under pressure” decision making.

Given the real-world situation, where human planners tend to be the only ones considered able to tackle such a problem, the innovation hereby suggested consists of an automated, artificial intelligence based, system capable of objectively driving the search and implementation of good solutions, without being influenced by pre-existing knowledge, mimicking a powerful lateral-thinking approach, so difficult to perform when management pressure impedes and daunting tasks bound the human rationality.

Ranking the effectiveness of a candidate solution, where path-dependency and all the complex effects may bias the final outcome, is not a matter trivially manageable by traditional operational research-style systems where no dynamics (recursive phenomena, feedbacks, non-linearity) appear. In order to break through the limitations that an analytical specification of the problem imposes, the Agent-Based Modelling paradigm had to be taken into consideration.

Thanks to ABM we enjoy the opportunity of “in-vitro” experimenting every imaginable scenario, by executing the planning in a virtual lab, where the production events happen instead of simplistically being computed. In this way we avoid following a reductionist approach, clumsily based on the usage of a static representation of the enterprise world, squashed into a cumbersome system of equations.

1 The problem

Matching tasks to units, under additional constraints, is the key issue. While certain constraints are to be regarded as ’hard’ (let’s think of a technical issue rendering some of the production units useless in working on particular a (sub-)task, thus reducing the set of available units), others are ’soft’ constraints: different units perform better on certain tasks, whereas others can suboptimally do, maybe with worse (yet acceptable) results, or take a longer time.

The sequencing of tasks is, on the other hand, one of the degrees of freedom of the problem, being the choice of giving priority to one task driven by timely delivery constraints.

For the sake of readability in this paper the words “order”, “task”, “job” will be used interchangeably.

---

*University of Turin, email: ferraris@econ.unito.it
†University of Turin and LABORatorio 'R. Revelli', email: matteo.morini@unito.it
1.1 Minimizing the production overall cost

Different production plans result in varying (aggregate) production costs. Each evaluation in terms of costs is made by adding several components: some of them are costs in a proper sense, others are more like abstract values by which we try to capture the economical impact of undesirable situations. Examples of the first kind are the setup costs; on the other hand delayed deliveries are certainly unwanted, even if not directly expressable as economical losses. Being considered an unreliable supplier because of repeated delays, in the long run, leads to unsatisfied customers being lost. This is, of course, an hardly economically quantifiable loss: it depends on how the firm’s management perceives the importance of reliability, and how strongly is feared the risk of losing a repeatedly “deluded” customer.

1.2 Textile technicalities explained

The simulation is performed on and limited to, for the sake of simplicity, one of the production chain tasks only: proper spinning. Previous and successive operations can be overlooked, since they normally take place in oversized departments. Warping and combing, for instance, require relatively inexpensive machinery to be completed: it is common practice to buy extra units ‘just in case’, since most of the plants value comes from spinners. The department where extreme care must be taken in avoiding any bottleneck effect is the spinning room.

We may confidently say that, should a good production plan be found for the spinning, the raw materials availability could be taken for granted, and the operations due to be performed up- and downwards the production chain could be arranged easily, not acting as constraints.

Finding a good production plan often implies dealing with mutually exclusive goals, in situations ridden with trade-offs. The only reasonable way to manage so many different aspects simultaneously is to reduce everything to its economical meaning, and it is hardly a straightforward task.

1.2.2 Timely delivery

Each order the firm is asked to produce is labeled with an “expected delivery date”: customers are promised their yarn will be ready to ship by an approved calendar date (sometimes stringent conditions are imposed by “big” buyers), which they expect to be reliable. Should the delivery constraints be missed, a disappointed customer would, to say the least, complain bitterly. We have a situation which is very difficult to express in economical terms; very seldom a penalty is contractually established, rather the firm reputation is at stake, and the risk is to lose customers.

In order to keep into account, besides of the setup constraint (“less is better”), this additional constraint, a figurative cost has been introduced. It consists of an amount of money associated with the delay and the importance, positively correlated with both: the longer the delay and the bigger the order, the higher the (not-so-metaphorical) cost to be
expressed in symbols:

$$DC_i = f(\tau_i^s, w_i^s)$$

where the delay cost DC for order $i$ grows as the delay $\tau$ and weight $w$ (in kilograms) grow.

It becomes clear that sequencing similar orders on
the same spinner is not an option: the freedom to
take them is at odd with the need to satisfy
the timely delivery condition. A simplified example
is presented (see Appendix, gantt sample).

1.2.3 Simultaneous setups, patrolling

To make things even worse (and almost impossible
to deal with “by hand”, which is nowadays the
only viable way available to enterprises) further
constraints are to be kept into account.

Production units setups, for instance, are performed
by specialized workers; the number of setup
2 teams available is limited, thus limiting the amount
of setup operations which can happen at the same
time. The effect of a missed setup (because of the
unavailability of a team) on the production is simply
a delay in the production of the order: no setup
2 can be performed until one of the busy setup teams
is available again. The total production time, and the
time the order will be ready to ship, will be determined
by the actual production time plus the initial
delay.

Other employees are committed to the so-called
spinners “patrolling”: they are required to follow
the ongoing production, ready to fix any problem
should occur. A patroller is normally assigned to 4 to 6
spinners to watch; the complication here arises from
heterogeneity in the behaviour of different spinners:
every different yarn features a specific likelihood
to create (generically speaking) problems, that is to
draw more or less attention from the patrollers. A
patroller will be able to follow productions that are
problematic up to a certain point: the average must
be kept below this critical point. Above the limit,
production times will grow (in a more or less foreseeable
way) for all of the spinners under the over-
loaded patroller.

An index of “problematiccy” is needed in order to
manage such a subtle issue. The patroller load $PL$
corresponds (for the $n$-th patroller) to the sum of the
“problematic index” $p$ for each order $i$ multiplied
by the number of heads, $h$, available on the spinner
$j$.

$$PL_n = \sum_{i=1}^{S} h_i p_j$$

Index $PL$ is normalized in order to have 1 as the
maximum tolerable patrolling load. Above this load,
orders production times increase by empirically
determined amounts:

<table>
<thead>
<tr>
<th>$PL$</th>
<th>$\Delta PT$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0 &lt; PL \leq 1$</td>
<td>0</td>
</tr>
<tr>
<td>$1 &lt; PL \leq 1.2$</td>
<td>+10%</td>
</tr>
<tr>
<td>$1.2 &lt; PL \leq 1.5$</td>
<td>+25%</td>
</tr>
<tr>
<td>$PL &gt; 1.5$</td>
<td>+25%</td>
</tr>
</tbody>
</table>

$PT = production\ time$
An accurate cost tracking and accounting is instrumental to a good final result: the figurative cost of each plan enters the solutions generator (the genetic algorithm), where it is used to evolve subsequent generations of solutions. Even small distortions may disrupt the search process towards inefficient regions of the solutions space, prolonging computational times and considerably worsening the quality and reliability produced solutions.

2 Experimenting solutions

The Enterprise Simulator is the module where solutions are experimented, that is where the simulation takes place. A model of the supply chain under scrutiny is used in order to watch candidate plans ‘happen’: the production process is represented in abstract, resorting to representative agents. Production units, agents, set-uppers agents, patrollers agents have been developed with the aim of giving simple yet exhaustive representations of their respective roles. Even production orders are embodied by dumb agents: objects encapsulating all the informations pertaining to the tasks to be performed, which are bounced between proper agents that act based on the informations they achieve from the orders themselves.

Presenting how the process takes place in the model is out of the scope of this paper; the steps - in a way absolutely adherent to the real process - implemented are: orders reception (in batch), orders dispatching to production units (filling queues), PUs setups, involving setup time computation after setup teams gathering, patrollers capacity reservation, production, repeat.

Ongoing and predetermined orders, already loaded on PUs and/or already due, are completed before initiating the candidate plan evaluation.

3 Inventing solutions (enters the Golem)

To search a good planning solution into the enormous set of possible ones, a Genetic Algorithm has been employed. The well known AI paradigm based upon the natural evolution process was invented by J. Holland (1975). The idea was to emulate the natural evolutionary process performing reproduction and death of structures that are representing a strategy. Provided that a whole set of structures is normally called the population of the GA, each of them is analogously named as an individual; each one encodes a strategy into a binary string called genome. After having created a random set of structures they will be evaluated, one at a time, by performing the strategy it is representing into an appropriate environment. In this way an evaluation of each structure becomes possible, so that each individual can be assigned a value measuring its goodness, the so called fitness of the individual. When the whole set has been evaluated, an evolution can be performed: each individual is assigned a probability to reproduce and a probability to die, according with its fitness value, i.e. high fitted genomes will have a high probability to reproduce and a low probability to die and vice versa. Reproduction is made by copying and crossing a couple of individual’s genomes to obtain a couple of new structures to put into two new individuals, these new born individuals will replace two old structures selected to die. By performing this algorithm in loop the population becomes more and more fitted and the better types tend to spread into the population. The GA method is very useful when a wide set of alternatives has to be explored: it is general-purpose, it does not require any previous coded knowledge about the problem and it allows finding reasonable solutions in a short time.

To face the planning problem a special, but general, implementation of a GA has been employed. The goal was to set up an empowered GA that could handle individuals compound by more than one structure and structures defined on a very large alphabet. Another requirement was that this special implementation of a GA, the Golem, needed to handle special structures where all the alphabet symbols appeared only once.

The decision to write a special GA was due to the particular topics of the problem. Each candidate strategy aimed to solve it can be split into two parts:

1. which machine will have to make an order
2. which priority will be assigned to each order

The two parts interact between each other in a complex way so the goodness of a solution depends on the goodness of each of them, but it is not possible to determine the contribution of each part to the performance of the solution. Both have to be evaluated simultaneously. Unless that, the contents of each part are very different and they could be coded in a highly different way. The first part could be expressed by a sequence of numbers, each of them identifies a unit, whereas the position of each code number identifies the order to be made. Adopting the same structure for the priorities the problem to
assign univocal values to each order has to be faced. In addition the code numbers are defined on a set which cardinality is given by the number of machines the enterprise owns, while the cardinality of the priority set is defined by the number of orders the enterprise is going to plan. Resorting to the standard two-symbols (0,1) alphabet would have caused an uneffective representation of the solutions space, given the problem to represent each number in binary code every time the number of orders, or the number of machines, is not a power of two.

The Golem tackles the aforementioned issues by allowing the user:

1. to decide independently for each genome how many symbols need to be used by the coding alphabet, i.e. how many different values will be used in it
2. to decide a different length for each genome, i.e. how many positions it will include
3. to handle genomes where each symbol of the related alphabet will appear only once.

In addition the Golem was written taking into consideration:

1. the robustness of the methods exposed to the user, who can hardly misuse them
2. the efficiency (performance-wise) of the program

The Golem features all the methods to handle and control the evolutionary process and allows the user to interact in a friendly way. The user has simply to define the structure of the individuals by coding the number of genomes and the parameters of each of them: type (univocal or random), length, alphabet cardinality. She then interacts with the Golem, asking for an individual to evaluate and, after having performed the evaluation, returning it to the Golem. When all the population’s individuals have been evaluated, the Golem automatically performs the evolution. The Golem code has been optimized to ensure a high performance level, either by controlling the user’s inputs and by paying a particular attention to reduce the computing time.

4 Experimenting invented solutions, ergo suggesting the good ones

The evolution process performed by the Golem is driven by evaluating each single candidate solution appearing in the GA population. The production plans require an estimation as accurate as possible, incorporating every element of the dynamic interaction characteristic of the enterprise operations. It is the existence of such relationships among the intervening parts which distinguishes the problem as one of a complex kind: the aggregated outcome differs from what is obtained by the single components.

Keeping in mind the facts mentioned above, the unfeasability of operating by decomposing the problem in parts is self-evident: the interactional effect would be totally missed; analogously, resorting to mathematical functions, static by their own nature, would imply neglecting all the time-related features, which are fundamental when it comes to plan actions intended to happen over time, being themselves subject to scheduling.

Computer simulation, by allowing management facts to happen in an artificial laboratory (the enterprise model), permits to quantify and express costs, whether figurate or not, generated by each candidate schedule, accurately and significantly, in order to promote the search for the best solution to the given problem.

By using the same instrument, conceived in order to perform what-if analyses, driven by human decisions, in order to rank the GA-made solutions, allows to compare directly what’s produced by the human heuristics versus what’s made by AI techniques. Apparently it’s the only viable method providing a shared metric which permits, given the amplitude of the problem, to decide whether the search direction is a productive one or not.

In order to exploit the enterprise simulation to these purposes, it is required that objects intended to act as a bridge between the (scheduling) plan from the inferential method (the Golem) to the entrepreneurial metric.

In designing the Golem, that concerning this activity is just one of the advantages aimed at: the chance to use an extended (symbolic) alphabet allows some coding issues that during the first trials performed by standard AGs slowed seriously the search process, to be solved. An alphabet restricted to binary digits forces production units and orders number to be expressed by grouped symbols (as many as needed in order to the maximum value in the definition domain to fit); wherever the defined domain is less dense than the natural numbers set (when dealing with orders classified by differentiating their number by thousands or tenths of thousands, for instance), several non-significant solutions may appear. Translation algorithms need to be employed, in such circumstances, which keep-
ing such facts into account, operate transformations (remapping) unknown to the AG; in the worst cases the same value gets assigned to formally different structures. Such behaviours can sensibly mislead the solutions learning and refinement process, keeping effective results from being efficiently achieved; execution times may dilate considerably.

A further issue emerged from the orders execution priorities. A standard GA in this case tended to produce non-univocal outcomes: the same priority may have been assigned to several different orders. Artificially differentiating equal values, based on the position within the structure for instance, might have impaired the GA abilities also in this situation. The system would have been kind of “deceived” by such artifacts. Providing the ability to opt between different operators, applicable to different kinds of genomes, the Golem could solve this issue too.

Achieving reasonable solutions quickly is fundamental to the enterprise: by analysing the experimental results a logarithmic trend of the solutions goodness have emerged clearly, functional to the number of evolutions performed. Practically speaking, the Golem is able to rapidly improve the solutions during the early stages of learning, while its productivity decreases as the optimum is approached. Going for population convergency appeared a suboptimal behaviour: halting the system after a certain number of evolutions seems way better than waiting for a long time in exchange for small improvements.

5 Conclusions

Production planning constitutes a typically complex problem: the interacting parts taking part in the process makes impossible the application of traditional search procedures, based for most part on the decomposability of the problem as a prerequisite.

Given an (although limited) number of tasks to schedule, even the plain enumeration of the possible solution becomes practically unfeasible, given the combinatorial explosion implied. In this scenario the limits of applying heuristics based on human experience have appeared: the human mind attempts to solve the problem operating on limited subsets at a time, implicitly decomposing the complex problem, thus missing an overall view on it. Every single decision taken on the assignment of a task onto a production unit constitutes a sensible “cut” on the solutions space, resulting in neglecting the exploration of large areas.

Applying GAs let us exploit their implicit parallelism, both from a computational and a reasoning standpoint: starting from randomly generated solutions, avoiding pre-digested strategies, the GA also considers solutions that would be rejected by a human solver as absurd ones; not seldom innovative ideas are found among such apparently suboptimal candidates, and they are the ones that give superior results.

Apparently this is the main reason for the superiority of the system with respect to the human approach. It demonstrates itself far superior both in computation duration - efficiency - and final results - efficacy.

The system put into place constitutes, though, just a starting point: ways to improve the efficiency are being investigated and experimented, by distributing the “thinking” part of the work, the simulation, on several distributed nodes of a computer network, drastically incrementing the execution parallelism. At the same time work is being done on making the inferential engine (the Golem) more powerful, by introducing even more dramatic variations with respect to the standard GA’s. The ongoing tests concern: clustered, cooperating GA’s, and GA’s featuring varying populations and variable-length individuals.

Glossary

**head** *(spinning mills)*: one of the (tenths to hundreds of) elements working on a single thread, constituting a spinning mill.

**card** *(id.)*: a toothed brush used to disentangle fibers

**rotor** *(id.)*: a rotating device used in transporting fibers

**nozzle** *(id.)*: a v-shaped element through which air flows

Appendix

- Gantt example:

Customers a and b demanded, respectively, for [A1, A2] and [B1, B2]. Orders A1 and A2 are, from a technical standpoint, similar, and require a negligible setup time between them. B1 and B2 are also very similar. Ignoring (by now) the delivery constraints the obvious plan is to sequence similar orders on the same spinner (solution 1):
<table>
<thead>
<tr>
<th>spinner #</th>
<th>t₀</th>
<th>t₁</th>
<th>...</th>
<th>tₙ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A1</td>
<td>q-A2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>B1</td>
<td>q-B2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The two customers, on the other hand, have different timing requests: 

- **a** needs A1 and A2 as soon as possible; 
- **b** is not pressing very much for a quick delivery and is fine for him to receive B1 and B2 by a later date. 

The most appropriate plan in this case would appear as follows (solution *i*):

<table>
<thead>
<tr>
<th>spinner #</th>
<th>t₀</th>
<th>t₁</th>
<th>...</th>
<th>tₙ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A1</td>
<td>l-B1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>A2</td>
<td>l-B2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The small letters preceding the second orders are meant to show the different setup times required in both situations: as expected, *q* stands for ’quick’ setup, *l* for ’long’ setup.

Even in an oversimplified situation like the one described above, the complicated management of incompatible constraints appears: what makes solution *i* preferable over *ii* are the actual setup and delivery “costs”, which must be accounted for as accurately as possible.

### References

<table>
<thead>
<tr>
<th>Reference</th>
<th>Title</th>
</tr>
</thead>
</table>


A TEXTILE SUPPLY CHAIN MANAGEMENT REQUIREMENT: IMPROVEMENT OF SALES FORECASTING

Sébastien Thomassey, Michel Happiètte, Jean-Marie Castelain.
GEMTEX-ENSAT
9 rue de l’Ermitage
BP 30329 59056 ROUBAIX cedex 1
France
E-mail: sebastien.thomassey@ensat.fr

KEYWORDS
sales forecasting, textile distribution, soft computing

ABSTRACT

Competition and globalization imply a very accurate production and sourcing management of the Textile-Apparel-Distribution network actors. A sales forecasting system is required to respond to the versatile textile market and the needs of the distributor. Nowadays, the existing forecasting models are generally unsuitable to the textile industry. We propose a forecasting system, which is composed of several models and performs forecasts for various horizons and at different sales aggregation levels. This system is based on soft computing techniques such as fuzzy logic, neural networks and evolutionary procedures, permitting the processing of uncertain data. Performances of our models are then evaluated using the real data from an important French textile distributor.

INTRODUCTION

Due to the competitive environment of the Textile-Apparel-Distribution industry, companies require a rigorous management of sourcing, production and distribution. Many transformations, which are required to produce textile item, and globalization, which causes dispersion of network actors, impose significant and not easily reducible manufacturing lead times. Thus, in order to deal with the customer's requests, companies often need to rely on an appropriate sales forecasting system to anticipate production volumes and to stock up items.

The features of the textile market activity also complicate the forecast procedure. Indeed, the proposed system must be able to:

- deal with a large number of items (about 15000 per year) and different aggregation levels of sales (figure 1),
- carry out mid-term forecasts (horizon of one season or year) to consider the quantities to produce and plan the sourcing, and short-term forecasts (horizon from 1 to 3 weeks) in order to re-plan the sourcing,
- use short histories and treat new items,
- take into account many factors, called explanatory variables, which influence sales (weather and calendar data, marketing action, promotions, fashion, economic environment, ...).

Figure 1: Example of textile items aggregation

The performance of classical time series forecasting models (exponential smoothing, Box and Jenkins model,...) (Franges 1999), depend strongly on the application field, the knowledge of the user and the forecast horizon (Dasgupta et al. 1994). However, due to the short historical textile sales data and the strong influences of explanatory variables, traditional models are generally unsuitable for textile sale forecasting. This suggests the need for new models to compensate for the limits of the traditional models.

In recent years, methods based on "soft computing" have been largely developed for various forecasting problems. Neural networks have been exploited by many forecasters (White 1992) (Zhang and Hu 1998) for their capacity to model non-linear relations, their training and adaptation facility. Fuzzy inference systems, which are able to model human knowledge and are tolerant of imprecise data (Zadeh 1994), can easily be interpreted and applied in forecasting...
(Klir and Yuan 1995). Taking into account the very strong and specific constraints relating to the industrial context in which we operate, we chose soft computing techniques to implement our sales forecasting models.

In the following section, we describe the general principle of our system, which consists of several models performing forecasts at various horizons (mid and short terms) and at different sales aggregation levels (family, item, size, and color).

The third section describes and evaluates our models with real data (3 years including about 42000 items organized in 322 families) supplied by an important French ready-to-wear distributor.

Finally, section 4 provides the conclusions.

**PROPOSED FORECASTING SYSTEM**

The proposed forecasting system is composed of several models (figure 2) which meet the various requirements of distributors (mid and short-term forecasts on aggregate data by family, item, size and color) according to features in the textile market (influence of explanatory variables, seasonality, reduced historical data).

![Global forecast system principle](image)

**Figure 2: Global forecast system principle**

**Notation**

The main used notations are the following:

- \( A \): historical season number available
- \( X : \) historical sales data of season \( N \) for family \( F, N \leq A \)
- \( X(t) : \) sales at last known period \( t \) of season \( A+1 \), for family \( F \)
- \( \hat{X}_{F+1} \): mid-term sales forecast of season \( A+1 \) for family \( F \)
- \( \hat{X}_{F+1}(t+1) \): short-term sales forecast at period \( t+1 \) of season \( A+1 \) for family \( F \)
- \( U : \) selected explanatory variables of season \( N \) for family \( F \)
- \( x : \) item historical sales data of season \( N \) for item \( i \)
- \( x^{i,T} : \) historical sales data by size \( T \) of season \( A+1 \) for item \( i \)
- \( x^{i,C} : \) last known sale by color \( C \) of season \( A+1 \) for item \( i \)
- \( \hat{x}_{F+1}^{i,T} \): mid-term sales forecast by size \( T \) of season \( A+1 \) for item \( i \)
- \( \hat{x}_{F+1}^{i,C} \): mid-term sales forecast by color \( C \) of season \( A+1 \) for item \( i \)
- \( \hat{x}_{F+1}(t+1) \): short-term sales forecast by color \( C \) at period \( t+1 \) of season \( A+1 \) for item \( i \)

**Family sales forecasting**

In order to treat the sales seasonal behavior and the influence of explanatory variables, forecasting models need complete historical data of several years. To obtain such as historical textile data, aggregation of each product is required.
Mid-term family sales forecasting: AHFCCX model
The proposed automatic model, called AHFCCX (Automatic Hybrid Forecasting model with Corrective Coefficient of eXplanatory variable influences), mainly consists in:
- extracting the influence of explanatory variables (EV) (price, data calendar) of the historical sales,
- applying a simple forecasting model based on sales seasonality and not taking into account the influence of explanatory variables,
- adding the influence of explanatory variables of the future season to forecasts obtained from the past.
The proposed AHFCCX model is based on an automatic learning of a Takagi-Sugeno FIS (Sugeno 1985)(Takagi and Sugeno 1985) from historical data. The structural (choice of the inference rules) and parametric adjustments are established by a genetic algorithm and a gradient based method respectively (Thomassey et al. 2002a).

Short-term family sales forecasting: SAMANFIS model
The purpose of the short-term forecasting model is not intended to model the influence of the explanatory variables and the seasonality of sales, but to readjust the mid-term forecasts \( \hat{X}_{f+1} \) of the last sales \( X_{f} \). In this context, adaptation and training capacities of neural techniques thus appear very useful. The main constraint is to model complex relationships with limited historical data. We developed the model SAMANFIS (Short-term forecasting model by Adjustment of Mid-term forecast Adaptive Neural Fuzzy Inference System) based on a neuro-fuzzy system ANFIS (Jang 1993) which translates a fuzzy inference system into a neural network.

The input's variables are composed of real sales \( (X_{f-1}(t)) \), the mid-term sale forecast (AHFCCX model) \( (\hat{X}_{f+1}(t)) \) of the last week and mid-term sales forecasting of the future week \( (\hat{X}_{f+1}(t+1)) \). The model output is the short-term forecast \( (\hat{X}_{f+1}(t+1)) \).

Item sales forecasting
Mid-term items sales forecasting: IDAC model
Generally, the items of a new collection do not have a historical data (95% of items are not renewed). The use of the mid-term forecasting AHFCCX model is unsuitable. The proposed model, called IDAC (Items forecasting model based on Distribution of Aggregated forecast and Classification) (Thomassey et al. 2002b) estimates the item (i) sales of the same family \( (F) \) without requiring historical data. This model is based on the family sales forecasts distribution (computed by model AHFCCX) from the life curves achieved by a classification procedure. This last method integrates the descriptive criteria of items which are available in the distributor database such as average prices, number of stores distributing the item or variety colors. The envisaged classification procedure is a hierarchical type, which consists to join individual items or clusters together sequentially.

Sales forecasting by color and size
The finer the data aggregation level is, the more difficult the forecast is. Difficulties encountered in the preceding section (limited historical data and references not renewed) are thus amplified, especially for forecasts by color.

Sales forecasting by color
Generally, the various colors of a collection are selected according to the fashion evolution and are not renewed from one season to another, except for the basic colors (white, black...). Sometimes, they are also added or withdrawn during the item life time. Due to these constraints, experts must intuitively estimate forecasts by color from their market knowledge. In this context, automation appears difficult. The system under consideration relies on the forecast distribution already established by item. The initial distribution is carried out by the expert (or if possible by the history), and the ongoing sales allow adjustment of forecasts using SAMANFIS model.

Sales forecasting by size
Contrary to forecasting difficulties associated with color, the forecasts by size can be estimated relatively easily by the distributor. The distribution of sizes remains steady from season to season; therefore, historical data can constitute a reliable information source. Thus, the proposed sales forecasting method is based on the corresponding items sales distribution, computed according to the historical data. The mid-term forecasts being generally reliable, the short-term readjustment according to the last sales is not necessary for the sizes.
EXPERIMENTATION

Performances of our models are evaluated with a real database from the past three years (1998, 1999 and 2000), which is composed of 322 families and 42000 items from a large French textile distributor. The first two years are employed for the training of the models' parameters and the third year evaluates the forecasts' accuracy.

Accuracy criteria

In order to quantify the errors of tested models, we selected RMSE and MdAPE criteria. The MdAPE criterion, only slightly sensitive to aberrant points, is recommended to compare models on numerous series (Armstrong and Collopy 1992).

Family sales forecasting

For comparison, we have also tested the following traditional models: naïve model (BMS), Holt Winter model with Seasonality (HWS), Automatic model Selection by the Forecast Pro software (AS-FF), ARMAX model.

Mid-term family sales forecasting

The AHFCCX model, tested for 322 families on a 52-week horizon, significantly improves the accuracy criteria RMSE and MdAPE compared to traditional models. We can also see that the ARMAX model is less efficient than AS-FF and HWS models in our simulation. Despite this model taking into account the explanatory variables, its linear structure is unable to map series with short historical data.

<table>
<thead>
<tr>
<th>Model</th>
<th>Aggregation</th>
<th>Horizon</th>
<th>number of sel</th>
<th>improvement of RMSE</th>
<th>improvement of MdAPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AHFCCX</td>
<td>family</td>
<td>52 weeks</td>
<td>322</td>
<td>+9.1 / HWS</td>
<td>+9.7% / HWS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+35.8% / ARMAX</td>
<td>+29.6% / ARMAX</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+15.7% / AS-FF</td>
<td>+21% / AS-FF</td>
</tr>
<tr>
<td>SAMANFIS</td>
<td>family</td>
<td>1 week</td>
<td>322</td>
<td>+3.2% / HWS</td>
<td>+13% / HWS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-13%/ARMAX</td>
<td>8% / ARMAX</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+5.2% / AHFCCX</td>
<td>+33% / AHFCCX</td>
</tr>
<tr>
<td>IDAC</td>
<td>item</td>
<td>52 weeks</td>
<td>4070</td>
<td>-4.6%/RB</td>
<td>+21% / RB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+1.2% / IDA</td>
<td>+25% / IDA</td>
</tr>
<tr>
<td>SAMANFIS</td>
<td>item</td>
<td>1 week</td>
<td>322</td>
<td>+55% / IDAC</td>
<td>+36% / IDAC</td>
</tr>
</tbody>
</table>

Short-term family sales forecasting

Three models, HWS, ARMAX and SAMANFIS tested for 322 families on a one-week horizon, ensure a reduction of the mid-term forecasting model (AHFCCX) error. The SAMANFIS model gives the best performance with the MdAPE criterion. However, the ARMAX model proves more precise according to the RMSE criterion.

Items sales forecasting

The three mid-term items sales forecasting models RB, IDA and IDAC, and short-term SAMANFIS model are tested on the 4070 items sales from families with the highest sales turnover. The model RB, which is used for new items without proper history, uniformly distributes the family forecasts by the following relationship:

\[ \hat{x}_{d,i}(t) = \frac{\hat{x}_{d,i}(t)}{N(t)} \]

where \(N(t)\) is the number of items which sold at period \(t\). IDA model [9] relies on the same sales forecasting distribution principle as IDAC model. The estimation of the sales profiles of new items results from the sales profile average of all historical items belonging to each family.

Mid-term item sales forecasting

The IDAC model classification procedure generates, for the same family, clusters quite distinctive according to item price and number of stores distributing the item. However on a 52-week horizon, the results are sometimes ambiguous. In fact, the IDAC forecasting model is often more accurate than either the RB or the IDA models, in particular the estimation of the item sale profiles; nevertheless, a deficient sales quantity forecast of some items is strongly sanctioned by RMSE criterion. Thus, the RB model obtains in average the best performance in term of RMSE, compared with the IDAC model.

Short-term item sales forecasting

The last sales of the season allows an important readjustment of the IDAC model forecasts on a one-week horizon, in particular the quantity (mid-term sale profile estimation are generally relatively correct). Thus, the short-term SAMANFIS forecasting model permits a significant improvement of the mid-term IDAC forecasting model: RMSE and MdAPE criteria decrease of 55% and 36% respectively. These results indicate the benefit to re-estimating the forecasts each week, especially on the item aggregation level where the mid-term forecast accuracy is sometimes uncertain.
Sales forecasting by size and color

Sales forecasting by size
Sales distribution on different sizes is relatively stable. In general, the sales forecast by size appears to be highly reliable and easy to evaluate.

Sales forecasting by color
The forecast by color is more complex to evaluate, in particular by the need to rely on estimates from the experts. It is generally very difficult to obtain these data within textile distribution. The tests are carried out on two women T-shirt items available in three basic colors (black, grey, white) and ten fashion colors (red / kaki, rust / red...).
The initial sales distribution on the item basic colors is relatively easy for an expert to determine and the SAMANFIS model ensures a precise distribution readjustment according to the latest sales. However, the sales distribution for fashion colors fluctuates much more. The expert estimates of the quantities distributed between the various colors is more difficult. The SAMANFIS Model also gives results which are more difficult to interpret and does not systematically improve the initial expert distribution when incorporating the sales.

CONCLUSION AND PROSPECTS
From real data provided by a major textile distributor, we have applied our global system which is composed of various forecasting models. These deal with sales forecast aggregated at different levels, for mid and short-term horizons. The use of real data forces the proposed models to take into account the strong constraints of the textile industry.
The mid-term family sales forecasting performed by a model, based on a fuzzy inference system (AHFCCX model), integrates separately and automatically the sales seasonality and the influence of explanatory variables. The principle of the IDAC model, which forecasts the mid-term sales items, relies on the family forecast distributions of the corresponding items and on the life curve estimation from a classification procedure.
The short-term family and item sales forecasting (SAMANFIS model), founded on a neuro-fuzzy procedure, readjusts the mid-term forecasts of the AHFCCX and IDAC models.
The textile items sales distribution by size is relatively stable. Thus, the forecast by size is computed using a basic historical distribution. The sales distribution on colors, which is very sensitive to fashion phenomena, requires expert intervention. This intuitive method, supplemented by the short-term SAMANFIS model readjustment, gives acceptable results for basic colors (white, grey, black) but is inadequate for fashion colors.

The main extend envisaged are the implementation of the short-term SAMANFIS forecasting model for a longer horizon (two to three weeks) sometimes required for the stores sourcing and the achievement of a more effective methods for the sales by color, by integrating the fashion phenomena.

REFERENCES

BIOGRAPHY
SÉBASTIEN THOMASSEY received the M.Sc. degree from the Ecole Nationale Supérieure des Arts et Industries Textiles (ENSAIT) of Roubaix in 1999 and the Ph.D. in automation from the University of Lille 1 (France) in 2002. He is assistant Professor in logistic and automation at the ENSAIT since 2003. His research interests include classification and sales forecasting of textile items.
A SIMULATION-BASED SCHEDULING SYSTEM AND ITS IMPLEMENTATION IN A TEXTILE OPERATION

Jeffrey A. Joines
Kristin A. Thoney
College of Textiles
North Carolina State University
Raleigh, NC 27695-8301, U.S.A.
Russell E. King
Thom J. Hodgson
Industrial Engineering
North Carolina State University
Raleigh, NC 27695-7906, U.S.A.

KEYWORDS
Scheduling, Production, Transportation, Discrete Simulation, Deterministic

ABSTRACT
Job shop scheduling is a complex problem that has been studied extensively. There are many different approaches that have been proposed to treat this problem. Several of these methodologies are discussed with respect to their potential use in industry. A simulation-based scheduler, the Virtual Factory, is presented that has proved to schedule well both in theory and practice. Its development and implementation in a textile operation are discussed. The scheduling system has allowed the company to reduce its cycle time by at least one week.

INTRODUCTION
As global competition has become more fierce, meeting customer due dates is increasingly important to the success of businesses. Developing good production schedules is an essential part of satisfying due dates, as well as achieving other production objectives. A great deal of research has been focused on solving the job-shop problem, over the last forty years, resulting in a wide variety of approaches. These approaches can be classified into five general categories: Industrial Practice, Math Programming, Rule-based systems, Heuristic/Computer-Search, and Simulation-Based. Industrial practice includes different scheduling philosophies like manufacturing resource planning (MRP), just-in-time (JIT) or pull systems. MRP assumes an infinite capacity where a job’s start time on a machine is equal to the due date minus the remaining processing times. However, in practice in highly constrained systems (i.e., capacity constrained), infinite capacity analysis cannot adequately schedule the shop floor. Applications of mathematical programming approaches include linear, nonlinear, dynamic, integer programming, and network flows. Computational complexity allows only very small problems to be solved to optimality in real (or near real) time, since solution time generally increases exponentially with problem size. This is especially problematic for job shops, where there may be thousands of jobs.

Often simple rules are used to prioritize the jobs on each of the machines like SPT (shortest processing time), LPT (longest processing time), due-date ordering, etc., depending on the criteria. However, these rules are myopic and, at best, optimize the individual machines or work centers rather than the global system. The Theory of Constraints (or TOC as it is called) is a relatively recent development in the practical aspect of making organizational decisions in situations in which constraints exist. Goldratt (1992) first described the theory in his novel, The Goal. In many organizations, TOC and TOC logic form major portions of the organization's philosophy of continuous improvement. However, TOC only identifies a static bottleneck. However, in most real cases, the bottleneck shifts to various portions of the shop.

Recently, much effort has been concentrated on hybrid methods to solve scheduling problems, as a single technique has not been able to solve this stubborn problem. As a result, much effort has recently been concentrated on techniques that combine myopic problem-specific methods and a meta-strategy, which guides the search out of local optima. These approaches include tabu search, hybrid genetic algorithms, etc. However, these methods still suffer from some of the problems. They cannot solve industrial-sized problems with tens of thousands of job operations, with hundreds of machine types each having multiple machines of each type, with workers being a constraint (i.e., more machines than workers), etc. In the past, simulation methodologies have been primarily used to plan and analyze systems. Over the last decade, simulation models have also been utilized as scheduling tools when "work to" schedules have been generated and then used on the shop floor to control the flow of product. Therefore, simulation-based scheduling systems offer the ability to accurately model the complexities of the current system (i.e., people, breaks, shifts, trucking times, etc.). Many of the current major simulation vendors offer scheduling modules. The goal of any system is to develop schedules very quickly such that decisions can be made and alternatives tested. The Virtual Factory (VF), developed at NC State, is one such tool that has been found to provide near-optimal solutions to industrial-sized problems in seconds (Hodgson et al., 1998, 2000). The VF is a simulation-based procedure, which solves deterministic problems by minimizing maximum lateness, $L_{max}$.
This paper will discuss the VF algorithm and the implementation of a VF derived scheduling methodology for a garment maker. Section 2 will describe the VF along with experimental results of the solution quality. In Section 3, the textile operation is described in detail. Section 4 will describe the scheduling system that has been developed and implemented, which is an extension of the academic VF. Section 5 presents the conclusions.

VIRTUAL FACTORY

The Virtual Factory is based on a conceptually simple idea first presented by Vepsäläinen and Morton (1988). The idea is essentially that slack fails to provide good schedules because it does not take the queueing time of jobs into account. By estimating this queueing and incorporating it into a revised slack measure, better performance to due dates can be achieved. Calculating this queueing time is not easy and using historical estimates is often insufficient because it does not necessarily account for the composition of jobs currently in the shop. To overcome this difficulty, an iterative procedure is used whereby the actual queueing time for each job at each machine is recorded during one iteration and used as the queueing estimate in the next iteration. More formally, let \( d_i \) be the due date of job \( i \), \( p_{ij} \) be the processing time of job \( i \) on machine \( j \), and \( q_{ij} \) be the queueing time of job \( i \) on machine \( j \). Then the slack and revised slack of job \( i \) on machine \( m \) are defined, respectively as follows:

\[
\text{Slack}_{m} = d_i - \sum_{j \in m^+} p_{ij} \quad \text{and} \quad \text{Slack}_{m} = d_i - \sum_{j \in m^+} p_{ij} - \sum_{j \in m^{++}} q_{ij},
\]

where \( m^+ \) is the set of all operations subsequent to machine \( m \) on job \( i \)'s routing and \( m^{++} \) is the set of all subsequent operations to machine \( m \) on the routing sheet for job \( i \), except the immediate subsequent operation.

To evaluate the quality of solutions, a lower bound is computed. This is accomplished by decomposing the job shop into individual one machine problems. If we let \( r_i \) be the release time of job \( i \), then the earliest start and latest finish for job \( i \) on machine \( m \) are defined, respectively, as follows:

\[
ES_{i,m} = r_i + \sum_{j \in m^-} p_{ij} \quad \text{and} \quad LF_{i,m} = d_i - \sum_{j \in m^+} p_{ij},
\]

where \( m^- \) is the set of all operations preceding machine \( m \) on job \( i \)'s routing sheet. The earliest start is an effective release time, while the latest finish is an effective due date for each job at each machine. Using the preemptive earliest due date rule provides a lower bound on each of the individual one machine problems. Taking the maximum of all these lower bounds provides a lower bound on the overall job shop problem. Comparing the \( L_{max} \) scheduling solution to the overall lower bound gives us the maximum by which the simulation solution could exceed the optimal.

During the first several iterations of the simulation, the schedules tend to become better as the queueing time estimates improve. After the first ten iterations or so, the solutions tend to bounce around. In general, the more iterations run, the better solution achieved. Thus, the usual termination criterion is to run the simulation for a fixed number of iterations or until the lower bound is achieved. The Virtual Factory has been shown to provide near optimal solutions to randomly generated and industrial furniture manufacturing problems in seconds. Furthermore, the computation time is roughly linear in problem size.

We have begun testing the VF in a rolling horizon fashion to better represent the way scheduling algorithms are used in industry (Thoney et al. 2002b, Cho et al. 2003). Traditionally, in the production literature, job shop scheduling algorithms are evaluated on randomly generated data by generating static problems, where all jobs are initially available, and the factory is run until it is empty. But this does not accurately reflect how job shop scheduling algorithms are used in industry. Plants usually contain many different orders, with new orders arriving as older ones are completed. Scheduling is often performed on some regular basis. The best schedule is implemented until the plant is rescheduled. Thus scheduling occurs on a rolling horizon basis. The definitions required to explain the algorithm, as well as the algorithm itself are presented:

- \( t \) - Current time in days
- \( c_j \) - Completion time of job \( j \)
- \( N \) - Total number of jobs
- \( T \) - Total horizon length in days
- \( w \) - Number of days in warm-up period
- \( i \) - Number of iteration

1. Initialize \( t = 0 \)
   1.1 If \( t = w + 1 \), compute \( LB \)
   1.2 Release jobs whose \( r_i = t \)
   1.3 Run the Virtual Factory \( i \) iterations
   1.4 Implement the first day of the best schedule
   1.5 \( t = t + 1 \)
   1.6 Continue from 1.1 until \( t = T \)
2. Run the remainder of the best schedule until all jobs are finished
3. Initialize \( j = 1 \)
   3.1 If \( c_j > w \), determine if job \( j \) is the \( L_{max} \) job
   3.2 \( j = j + 1 \)
   3.3 Continue from 3.1 until \( j = N \)

Step 1 releases the new jobs into the system, runs the VF, and implements the first day of the best schedule. This procedure is repeated each day until the total number of days is reached. In step 2, the best schedule is run until all jobs are finished. In step 3, the latency for each job completed after the warm-up period is compared to the current maximum latency.

Figure 1 shows the performance of the academic VF on a 1 factory randomly generated rolling horizon problem.
with 50 machines where 151 jobs were released each day. This number of jobs was determined to balance factory input with factory output. Each day in the total horizon of 100 days, the scheduling algorithm was run for 100 iterations and statistics were computed based on all jobs completed after the initial warm-up period of 10 days. Each point on the graph represents the average of 20 replications. The due date range has been varied since this is known to effect the quality of scheduling solutions.

![Graph showing the operation problem base case](image)

Figure 1: Virtual Factory Performance

Recall that \( L_{\text{max}} \)-LB is the maximum by which the simulation solution could exceed the optimal solution. A positive difference between \( L_{\text{max}} \) and LB could be the result of a non-optimal schedule, a weak LB, or a combination of the two. The differences in Figure 1 are quite small considering that 90 days of factory performance was included in these statistics, with the latenesses of over \((90)(151)=13,590\) jobs taken into account. This indicates that the scheduling procedure is performing well. Similar results have been found when parameters are varied.

Many other extensions of the academic VF have been pursued to add industrial capability. These ideas have come primarily from our experience in furniture and textile manufacturing. We have pursued algorithms to minimize work in process while satisfying due dates (Zozom et al. 2003), investigated alternative routings (Weintrub et al., 1999), added transportation capability to schedule multiple factories simultaneously along the supply chain (Thoney et al. 2002a), and incorporated sequence dependent setup machines with a simplified setup structure (Taner et al. 2003). A version of the VF has also been used to help solve a military deployment problem.

**TEXTILE OPERATIONS**

The textile operation in which the VF was implemented is a cutting operation that takes fabric and cuts it into garment pieces and then ships sew kits out to various places in South America and the Caribbean to be sewn together into garments. This operation consists of two plants located approximately 30 minutes apart by truck. The operation has only five major processes (see Figure 2), but is nonetheless quite complex. The processing begins with orders arriving from the production planning system on a weekly basis. About 80% of the orders are received on Sunday, while the remaining orders come in during the week, potentially forcing rescheduling every day.

Those orders are then exploded into individual garments and ultimately the parts that make up those garments. During the next process, certain parts are assigned to like material groups. Then marker planning assigns these groups into spread groups and develops the markers to be cut. Multiple parts from different orders may be grouped together on a single spread, in an effort to increase productivity and material usage in cutting.

![Diagram of job shop scenario](image)

Figure 2: Job Shop Scenario

Spreading is the first major process, which consists of spreading out the material (rolls of fabric) to allow the parts to be cut out. Due to material type and width constraints, certain types of fabrics can only be processed on specific spreaders. The next process is the actual cutting of the material, which can take place on automatic Gerber cutters or die cutters. Capacitated queues exist between the spreaders and the cutters. The spreading and cutting of the material takes one week in the current situation. Once the parts are cut, they can be sent either to lotting along with lace and die cut parts or to molding. Both the spreading and cutting operations are group operations (i.e., parts are processed together in batch before they can be sent to the next operation). Some of the parts in certain garments require a molding process, which is performed at the second plant. The molding operation is one of the most complicated processes owing to the constraints. There are tooling restrictions based on the actual molds and the sheer number of molds of a given type and size. The operation also has sequence dependent setup times to switch out the tools. Different sizes of the same type require a slight setup (20 minutes), while different types require a longer setup (40 minutes), and changing from a dark color to a light color requires a setup of 10 minutes. Once parts leave the molding operation, they proceed to either lotting or another
die cutting operation. It takes approximately one week for parts to be molded and returned back to the original plant.

The final major operation, lotting, occurs in the original plant. In lotting, all of the parts for a garment are assembled into sew kits. This processing is not started until all of the parts of a kit are available (i.e., no partial lotting is allowed). Therefore, garments with molding operations take three weeks, while the non-molded garments take only two weeks to process.

In the current situation, each process was being scheduled locally (i.e., each process was developing a schedule that optimized its process) in weekly time buckets. The local scheduling was based on many different criteria. For example, the molding supervisor scheduled the jobs in her weekly bucket to both minimize setups as well as schedule easy jobs on Friday, while lotting always processed the jobs in longest processing time order. In an effort to cut the overall finished good lead-time, management decided to reduce the cycle time to process the garments in these two plants by at least one week. The facility had a good track record of shipping the garments out in the current cycle time based on the weekly buckets. However, the facility could not continue to schedule based on the current local criteria and be able to ship the garments out on time to meet a tightened cycle time.

This meant that the facility had to schedule the garment parts more efficiently in a global fashion to be able to meet the reduce cycle time. All of the processes needed to be linked together tightly with an overall global criteria and a common scheduling methodology. Priority lists needed to be generated for each machine that adhered to any restrictions due to sequence dependent setups, tooling, worker considerations, trucking between plants, grouping of parts into spreads, etc.

MODEL DEVELOPMENT

A commercial version of the VF was developed to simulate the textile operation. The commercial VF is an object-oriented simulation written in C++ and linked to a Visual Basic.Net interface with MS Access as the underlying database. (The academic version consists of just the simulation portion and is also written in C++.) The goal of the commercial VF implementation was for the engine (simulation-based scheduling system) to be independent of any scenario (i.e., data would drive the simulation). The interface was created so that it could easily be tailored to fit the current scenario that is being scheduled.

Simulation Engine

Several additional features needed to be incorporated into the commercial version of the VF that were not present in the academic version. These include tooling, restrictions that jobs may be processed on only certain machines of the same type, and sequence dependent setups. Although a simplified setup structure had previously been researched, the textile operation’s setups did not follow that structure. Because our scheduling is simulation-based, these changes were relatively easy to incorporate, easier than they may have been had we used a different type of scheduling.

The following algorithm was used to schedule the plants. Note that the way in which the commercial VF was used closely parallels the rolling horizon version of the academic VF when the commercial VF is run each day.

1. Download the current status of the plant including number of machines, number of workers, new parts and the remaining operations of the parts already in process.
2. Dynamically create the simulation using the current download by creating machines, resource groups, parts, events, etc.
3. Initialize the simulation to the starting condition (i.e., all machines are idle, workers have been removed, and parts are at the proper machines).
4. For the first iteration, simulate the system until all parts have been finished by ordering the queues at the various machines by increasing slack. Also record the queuing time that occurs for each part at each operation as well as the maximum lateness for all parts.
5. Repeat steps three and four for a specified number of iterations. However, for subsequent iterations, order the parts by using the revised slack calculation and the queuing time recorded in the previous iteration.
6. Using the queuing times from the iteration which had the minimum maximum lateness, rerun the simulation recording statistics (i.e., priority lists of every machine, the lateness of the individual of the orders, etc.).

Interface

Once the engine was designed and working, a visual interface needed to be developed to operate the engine that was built. The data that drives the engine can be divided into two major groups: static/semi-static and dynamic data. The interface maintains and manages all of the static data like the machine center types, the number of machines of each type, resource functional groups and the semi-static data like shift information (number of workers, beginning and ending times, etc.) and tooling information. This type of data is not changed very often and therefore is maintained locally. We consider the download of parts, part operations, and orders to be dynamic since it changes daily and is not maintained locally. Instead, it is imported into the local database from a separate data source. Therefore, the engine is independent of the data source. If the data source is changed, then the import facility is modified to accommodate the changes, but the engine remains intact. The interface provides the ability to modify the static data, import the dynamic data, run the scheduler, and view output statistics, priority lists, and critical graphs. The interface has the ability to allow the company to
modify and create new output reports. Figures 3 and 4 show an example of a priority list for a lotting station.

Implementation Issues

The simulation-based scheduling system has been implemented and is currently in use by the garment maker. Lessons learned include that data accuracy is a huge issue due to the fact that the data had never been used to drive anything as detailed as a scheduling system. As the key processes and the data were described, certain assumptions were made in the model that turned out to be wrong. Most of these dealt with the data download. The actual interface seemed to be more critical to the company than the quality of the schedules. Appropriate types of graphs, reports, and data management had to be put into place before the system gained acceptance.

Second, there had to be “buy in” from all levels (management, supervisors, workers, etc). After the engine began generating actual schedules, many supervisors and workers had difficulty believing the schedules and did not follow the dispatching lists. Instead, they spent time modifying the schedules. Luckily, upper level management believed that their employees should follow a priority list and convinced them to follow the lists exactly until something went wrong. After a couple of weeks, confidence in the schedules grew, and the employees became so dependent on the dispatching lists that when they did not have them, they complained.

CONCLUSIONS

Many organizations have implemented ERP systems to satisfy finance and production management at the top-level, but ERP systems are limited in meeting the needs of the shop floor. ERP systems are not very useful by themselves to react quickly with alternative planning and scheduling decisions when changes arise. On the other hand, simulation-based scheduling systems, like the Virtual Factory, can link to ERP systems, accessing current information on sales orders, production processes, bill of materials, and inventory data, to help make scheduling decisions that work on the shop floor. By capturing operational rules, finite capacity constraints, and management objectives, a simulation-based production planning system executes the plan through plant processes to quickly generate an efficient production schedule!

REFERENCES


TEXTILE PENETRATION PROCESSES
MODELING THE TRANSPORT OF ORGANIC SOLVENTS THROUGH PROTECTIVE CLOTHING

Juan Hinestroza  
College of Textiles  
NC State University  
Raleigh, NC 27695  
jphinest@ncsu.edu

Daniel De Kee  
Dept. Chemical Eng.  
Tulane University  
New Orleans, LA 70118  
ddekee@tulane.edu

Pratap Puri  
Dept. Mathematics  
Univ. of New Orleans  
New Orleans, LA 70148  
ppuri@math.uno.edu

KEYWORDS:
Mass transfer, mechanical deformation, diffusion, enhanced transport, facilitated transport

ABSTRACT
A mathematical model that modifies Fick’s Laws of diffusion by including a stress related diffusion coefficient was developed to simulate the effect of external mechanical deformation on the barrier properties of protective clothing. The model uses a linear viscoelasticity approach to account for the effect of stress on the transport of organic solvents through elongated materials. The predictions of the mathematical model were found to be in quantitative agreement with experimental data and illustrated the presence of stress hindered transport.

INTRODUCTION
In recent years, due to an increasing awareness of the risks associated with exposure to toxic and harmful chemicals in the environment, there has been increased attention to the development and testing of protective clothing. In addition to provide a permeation barrier to challenging chemical agents, these materials are required to maintain their structural integrity against mechanical stresses and excessive dimensional changes. For example, in protective clothing, gloves, boots as well as the areas surrounding the elbows and knees, face a potential decrease in protection, as a result of pronounced swelling and substantial deformation, that can lead to loss in mechanical and barrier properties and ultimate failure of the material.

It has been recognized that stresses, internal (swelling) and external (imposed mechanical deformation), play a major role in the diffusion and sorption of organic molecules through polymeric materials. However, studies on the effect of stress in transport of organic compounds have been limited to simplified theoretical treatments, and only in a few cases have experimental techniques been reported. It is accepted that the effect of an externally applied stress on the permeation resistance of protective polymeric materials is still not well-understood, due in part to the difficulty in realistically simulating the simultaneous presence of mechanical deformation and permeation.

THEORETICAL ANALYSIS
A concept originally proposed by Edwards and Cohen was used as the starting point for modeling the transport of organic solvents through polymeric materials under mechanical deformation. This model considers the transport of the penetrants as driven by both molecular diffusion and by a stress associated with swelling and external mechanical deformation. A governing equation for diffusion, considering the influence of stress and concentration is then expressed as

$$\frac{\partial c}{\partial t} = -D \nabla^2 c + E \nabla^2 \sigma$$

(1)

where \(c\) represents concentration, \(t\) stands for time, \(D\) is the molecular diffusion coefficient which is considered to be independent of concentration and \(E\) is a stress-driven diffusion coefficient. The terms on the right side of Equation (1) are associated with molecular transport and with stress respectively. Note that if \(E = 0\), Equation (1) reduces to Fick’s second law.

In order to account for the evolution of stress as a function of concentration and time a second differential equation is required. Inspired by linear viscoelasticity theory, a Jeffreys' type model was modified as follows

$$\frac{\partial \sigma}{\partial t} + \beta \sigma = v_1 \frac{\partial c}{\partial t} + v_2 \frac{\partial^2 c}{\partial t^2}$$

(2)

In Equation (2), \(\beta\) is a constant that can be considered to be an inverse of a relaxation time, and \(v_1\) and \(v_2\) are model parameters. For membranes in the glassy state, \(\beta\) is small and can be neglected, but if the membranes are in a rubbery state, \(\beta\) is not negligible. Note that for \(v_2 = 0\),
Equation (2) reduces to a Maxwell type equation. It is assumed that diffusion occurs in one direction (x direction) and that at \( x = 0 \) the concentration of the liquid chemical in the membrane equals the saturation concentration \( C_o \). The concentration of the chemical at \( x = L \) is assumed to be negligible. This assumption is supported by the fact that in the experimental setup used in this project nitrogen is continuously flowing through the lower permeation cell, sweeping away any chemical that reaches the surface. In order to facilitate the solution of Equations (1 and 2), the following dimensionless quantities are introduced

\[
c^* = c / C_o, \; x^* = x / L, \; \sigma^* = \sigma / (v_i C_o), \; t^* = t / \beta, \; \gamma_1 = \frac{D}{L^2 \beta}.
\]

\[
\gamma_2 = \frac{E v_i}{L^2 \beta}, \; \gamma_3 = \frac{v_i \beta}{v_i}
\]

Equations (2 and 3) in dimensionless form become

\[
\begin{align*}
\frac{\partial c^*}{\partial t^*} & = \gamma_1 \frac{\partial^2 c^*}{\partial x^*^2} + \gamma_2 \frac{\partial^2 \sigma^*}{\partial x^*^2} \\
\frac{\partial \sigma^*}{\partial t^*} + \sigma^* & = \frac{\partial c^*}{\partial t^*} + \gamma_3 \frac{\partial^2 c^*}{\partial t^*^2}
\end{align*}
\]

(3)

(4)

With the following boundary conditions

\[
c^*(x^*, 0) = 0
\]

(5)

\[
c^*(0, t^*) = 1
\]

(6)

\[
c^*(l, t^*) = 0
\]

(7)

\[
\sigma^*(x^*, 0) = 0
\]

(9)

Equations (3) and (4) are solved by using Fourier transforms and the final expression for flux versus time is then expressed as

\[
F(t) = \frac{-1 + \sum_{s=1}^{\infty} e^{-\lambda_s t} \left[ \frac{2}{\pi} \left( \frac{1}{1 + \gamma_3 \left( \lambda_s \lambda_2 - \lambda_1 \right)} \right) \right]}{(1 + s^2 \pi^2)^{\frac{1}{2}}} (\cos b t + i \sin b t)
\]

(10)

Where \( \lambda_i \) is

\[
\lambda_i = \frac{W \pm \sqrt{W^2 - 4s^2 \pi^2 \gamma_i V}}{2V}
\]

(11)

Where \( V = 1 + s^2 \pi^2 \gamma_2 \gamma_3 \) and \( W = 1 + s^2 \pi^2 (\gamma_1 + \gamma_2) \)

For the case where parameters \( \lambda_1 \) and \( \lambda_2 \) are real numbers

Equation (10) represents the final expression for the flux through a membrane as a function of time. However, in the case where the parameters \( \lambda_1 \) and \( \lambda_2 \) become imaginary numbers, Equation (10) still holds, but some adjustments must be made to facilitate the numerical computations as follows.

When \( \lambda_1 \) and \( \lambda_2 \) are complex conjugates that can be represented by

\[
\lambda_i = a + bi
\]

(12)

The final expression for flux becomes

\[
F(t) = \left\{ \begin{array}{ll}
-1 + & \\
(1 + s^2 \pi^2)^{\frac{1}{2}} & \\
\sum_{s=-1}^{\infty} & -2 \left( \frac{1}{\pi} \left( \frac{1}{1 + \gamma_3 \left( \lambda_1 \lambda_2 - \lambda_2 \right)} \right) \right) & \\
\sum_{s=1}^{\infty} e^{-\lambda_s t} & \\
\left[ \frac{2}{\pi} \left( \frac{1}{1 + \gamma_3 \left( \lambda_s \lambda_2 - \lambda_1 \right)} \right) \right] & \\
(1 + s^2 \pi^2)^{\frac{1}{2}} & \\
e^{-\lambda_1 t} & \\
e^{-\lambda_2 t} & \\
\end{array} \right.
\]

(13)

where

\[
c = \frac{-b + 2ab \gamma_1 - a^2 \gamma_1 - b^2 \gamma_2}{b \pi^2 (1 - 2a \gamma_3 + a^2 \gamma_3 + b^2 \gamma_3)}
\]

and

\[
d = \frac{-a + a^2 + b^2 + a^2 \gamma_2 - a^3 \gamma_3 - b^3 \gamma_3 - ab \gamma_3}{b \pi^2 (1 - 2a \gamma_3 + a^2 \gamma_3 + b^2 \gamma_3)}
\]

Equation (13) shows that even though \( \lambda_1 \) and \( \lambda_2 \) are complex numbers, the final expression for the flux is real. The final flux-time profile is predicted via Equation (10) or Equation (13) in terms of three parameters \( \gamma_1 \), \( \gamma_2 \) and \( \gamma_3 \). The values of these parameters provide information regarding the effect of deformation on the diffusion of a chemical through a polymeric membrane. In particular, the ratio between parameters \( \gamma_1 \) and \( \gamma_2 \) can be used as a quantitative relationship between the molecular diffusion coefficient \( D \) and the elastic diffusion coefficient \( E \) indicating whether the transport process is diffusion or stress controlled.
MATERIALS

Two different types of rubber were evaluated in this study. All the specimens were provided by the Defense Research Establishment (DRES) located in Suffield, Canada. Both natural rubber (NR) and Bromo-isobutylene rubber (BIIR) membranes were supplied into square sheets of 12 x 12 cm. and were used without any further treatment. These membranes are of interest in protective clothing applications and several batches have been previously characterized and studied\(^6,12\). The two chemicals chosen for this study were acetone and hexane. Reagent grade hexane was obtained from Aldrich Chemicals (Milwaukee, WI) while reagent grade acetone was purchased from Fluka Chemicals (Milwaukee, WI). These reagents were used without further purification.

EXPERIMENTAL METHODS

The equipment used for performing the permeation experiments has been described previously\(^1\). The major components of the equipment are an ASTM F-739 permeation cell, a custom-built biaxial membrane stretching device, and a Shimadzu GC-17A.

For tests under uniaxial deformation, the membrane was clamped between two of the mobile heads of the stretching apparatus and elongated in one direction. For biaxial stretching, the membrane was initially elongated in one direction, as in the uniaxial case, then it was clamped to the remaining mobile heads and stretched in a direction perpendicular to the initial elongation. After the membrane was deformed, the upper and lower chambers of the permeation cell were aligned and clamped into place using two Teflon gaskets and the specimen as seals. The liquid organic mixture was introduced into the upper (challenging) chamber of the permeation cell. Contact of the membrane with the organic chemical designated zero time for a permeation experiment. Periodically, over the next 8 hours, the downstream nitrogen sweep gas was analyzed for concentration of organic(s) by the gas chromatograph. To ensure meaningful comparisons of permeation data, a new membrane was used in each experiment. The apparatus was tested extensively and a maximum variation of 7% in the steady state permeation values was observed between membranes of the same batch.

Permeation experiments with acetone and hexane as well as mixtures thereof through NR and BIIR sheets at several degrees of elongation were performed. Mixtures of 66, 50, and 33 vol % at 298 K of each component were tested through membranes elongated uniaxially 20% and 40% as well as 20x20%, and 40x40% in the biaxial mode.

The operation parameters for the gas chromatograph were optimized in order to produce an analytical method capable of resolving the peaks of acetone and hexane in less than 1 minute. This fast detection was necessary to determine sudden changes in permeability as reported previously\(^5\). The column was a 30 meter capillary column coated with Supelco Wax® 10. A flow rate of 45 cm\(^3\)/min of nitrogen was used as the carrier gas. A total of fifty experimental conditions were tested; five different elongations, five different concentrations and two rubber materials following a full factorial experimental design.

MATHEMATICAL MODEL

\[
\text{Figure 1. Flux versus time for the permeation of hexane through a natural rubber membrane at various degrees of mechanical deformation: } \square \text{ 0\% elongation, } \blacktriangle \text{ 20\% elongation, } \bullet \text{ 40\% elongation, } \ldots \text{ Prediction of Equations (10) and (13).}
\]

The mathematical model, Equations (10) and (13), was used to analyze the permeation of acetone and hexane through NR and BIIR membranes under mechanical deformation. Since the model is based on continuum mechanics principles and does not account for more than one permeant at the time, only the permeation of single components was modeled. A three-dimensional search algorithm was set to determine the best fitting parameters \(\gamma_1, \gamma_2, \gamma_3\) and to minimize the least square of the error between the experimental value and that predicted by Equations (10) and (13). The important parameters to be analyzed were \(\gamma_1\) and \(\gamma_2\).

Table 1 reports the values for the best fitting parameter \(\gamma_2\) for the permeation of acetone and hexane through NR and BIIR membranes, and shows that the value of \(\gamma_2\) increased as the elongation of the membranes increased. Experimental data as well as best-fitting curves based on Equations (10) and (13) are shown in Figure 1 for permeation of hexane through NR. It can be observed that the model predictions are quantitatively correct. The values of \(\gamma_2\), the parameter related to the stress-driven diffusion coefficient, were found to be inversely proportional to the increase in steady permeation rate confirming the existence of stress-hindered transport.
Table 1. Values for fitting parameter $\gamma$ in Equations (10) and (13) for the permeation of acetone and hexane through NR and BIIR membranes.

<table>
<thead>
<tr>
<th>Degree of Elongation</th>
<th>Hexane - NR ($x10^8$)</th>
<th>Acetone - NR ($x10^8$)</th>
<th>Hexane - BIIR ($x10^4$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>20%</td>
<td>0.50</td>
<td>0.05</td>
<td>0.30</td>
</tr>
<tr>
<td>40%</td>
<td>1.00</td>
<td>0.08</td>
<td>0.40</td>
</tr>
<tr>
<td>20% x 20%</td>
<td>1.00</td>
<td>0.08</td>
<td>0.40</td>
</tr>
<tr>
<td>40% x 40%</td>
<td>8.00</td>
<td>0.10</td>
<td>0.40</td>
</tr>
</tbody>
</table>

CONCLUSIONS

The predictions of the mathematical model were in quantitative agreement with experimental data for the permeation of organic solvents through protective clothing materials. The values for the fitting parameters illustrated the presence of stress hindered transport. The model provided quantitative agreement for both breakthrough times and steady state permeation rates.

REFERENCES


BIOGRAPHY

Juan P. Hinstroza is an Assistant Professor in the department of Textile Engineering Chemistry and Science at North Carolina State University. Dr. Hinstroza comes from TIMES (Tulane-NASA Institute for Macromolecular Engineering and Science) where he pursued postdoctoral studies after obtaining his Ph.D. in Chemical Engineering from Tulane University in May 2002. His doctoral research work was focused on the effect of mechanical deformation on the barrier properties of polymeric materials and was funded by the U.S. Department of Defense and Department of Energy. Prior to graduate school, Dr. Hinstroza was a process control engineer for The Dow Chemical Company.

His research interests are in the general area of protective clothing against chemical and biological agents. His research work is devoted to the synthesis and development of novel barrier materials including permselective block copolymers, inorganic-organic nanocomposites, and electrostatic nanolayer assemblies. Dr. Hinstroza also works in the mathematical modeling of transport phenomena in complex protective clothing ensembles.
NUMERICAL STUDY OF FLUID FLOW IN PACKAGE DYEING

R Shamey, North Carolina State University, Raleigh, NC 27695-8301, USA
X Zhao, Heriot-Watt University, Galashiels, TD1 3HF, UK

KEY WORDS

Numerical Simulation, Fluid Dynamics, Velocity Distribution, Package Dyeing

ABSTRACT

The present study is based on a comprehensive analysis of flow properties in package dyeing process by coupling the Navier-Stokes equations with Brinkman equations. Navier-Stokes equations are also combined with Darcy’s law to be used as reference. The results of simulation illustrate the velocity and pressure distribution of the flow through the package (both in the tube and yarn assembly). The influence of shear stress on flow within the package is also discussed.

1. INTRODUCTION

Levelness of dyeing largely depends on the uniformity of distribution of liquor in the packages [1]. In order to consider the effect of convective dispersion of dye during the dyeing on levelness, the velocity of the fluid at any point and any time is required. However, measurement of flow at different parts of porous packages of various shapes is practically impossible. In addition, the solutions of mathematical equations governing fluid flow in porous media of complex shapes are very difficult.

Fluid flow in practical dyeing processes including package dyeing is assumed as streamline in nature and often described by Darcy’s law [2-5]. Despite the fact that there is no theoretical evidence to prove this behaviour, researchers claim that model predictions generally show good agreement with experimental data.

Schaf et al. [5] used the Navier-Stokes equations to simulate turbulent incompressible liquids. He assumed that flow through bobbins was linear and described the yarn package by a porosity model. The simulation results provide information on the static pressure and velocity distribution at every part of the dyeing vessel. However, due to the restricted assumptions used in this model, it may only be used for a narrow range of dyeing situations.

During the dyeing, the dye liquor is pumped through the tube and the yarn assembly of the package, which consists of a three-dimensional shape with the interstitial space filled with the dye liquor.

In most package dyeing machinery (Fig 1), the flow is from the (vertical) tube to the yarn assembly (parallel). It remains unclear whether the global transport of momentum by shear stresses in the fluid is negligible. In addition, yarns are soft and capable of undergoing substantial deformation. When packages have low density, their deformation especially at the surface of the package becomes more noticeable which is due to the higher flow rate and flow direction change that occur at this soft interface. Therefore flow characteristics through this type of porous medium will be different from those in a rigid porous medium such as sandstone, in which, flow is described by Darcy’s law. Darcy’s law assumes that the only driving force for flow in a porous media is the pressure gradient, and the global transport of momentum by shear stresses in the fluid is negligible.

![Flow through the package](image)

Figure 1: Flow through the package

Therefore, in order to investigate the dyeing process, it is important to obtain a better understanding of flow through the yarn package, and the influence of shear stresses on flow properties.

2. FUNDAMENTAL THEORY

The coupling of free media flow with porous media flow is very common in the field of chemical
engineering. The most common way to deal with free and porous media flow in a system is to couple Darcy’s law, which does not account for viscous effects, with the Navier-Stokes equations [6]. However, depending on the properties of the porous media and the fluid, it is not always appropriate to neglect viscous effects. The Brinkman equations [7] account for momentum transport through viscous effects and through pressure gradients in porous media and can be considered an extension of Darcy’s Law. The coupling between Navier-Stokes and the Brinkman equations is therefore highly applicable in the fields of chemical engineering.

Steady flow of an incompressible Newtonian fluid through a rigid porous medium is usually described by Darcy’s law as shown below [8]:

\[ u = \frac{k}{\eta} \nabla p \]  

(1)

Where, \( k \) denotes the permeability of the porous media, \( \eta \) the fluid viscosity, \( p \) the pressure and \( u \) the superficial flow velocity of the fluid. The permeability, \( k \), taken as a constant, is an intrinsic property of the porous medium. The velocity in Darcy’s law, \( u \), is actually a mean velocity over the cross section.

The Brinkman equations [7] describe the flow in porous media in cases where the transport of momentum by shear stresses in the fluid is not ignored. The model extends Darcy’s law to include a term that accounts for the viscous transport in the momentum balance and introduces the velocities in the spatial directions as dependent variables.

\[ u = \frac{k}{\eta} \nabla p + \eta \nabla^2 u \]  

(2)

Brinkman equation has the advantage of considering both viscous drag along the walls and Darcy effects within the porous medium itself.

The Navier-Stokes equations are the basic equations describing movement in incompressible fluids. They are derived simply by assuming that the impulse of the fluid must be conserved unless forces are acting on the fluid.

For the steady-state, incompressible flow, the equations can be written as:

\[-\nabla \cdot (\eta (\nabla u + (\nabla u)^T)) + \rho (u \cdot \nabla)u + \nabla p = F \]  

(3)

\[ \nabla \cdot u = 0 \]  

(4)

The first term is the rate of momentum gain by viscous transfer, the second by convection and the third by pressure forces. The constant \( \eta \) denotes the dynamic viscosity (kg m\(^{-1}\)s\(^{-1}\)), \( u \) notes the velocity vector (ms\(^{-1}\)), \( \rho \) is the density (kg m\(^{-3}\)), and \( p \) denotes the pressure (Pa). The \( F \) term is a source term representing external forces per unit volume.

In this study the Navier-Stokes equation has been chosen to model the flow of dye liquor through the tube of the package. The Brinkman equation and Darcy’s law have been chosen to model the flow of dye liquor through a bounded permeable yarn assembly. Combined with assumptions and proper boundary conditions based on the system specified, the 2D models were developed to investigate the flow property of package dyeing process, and the influence of shear stresses on the velocity distributions through the package.

3. SOLUTION AND DISCUSSION

3.1 Basic assumptions and system geometry

1) The density of dye liquor is considered constant under isothermal conditions.
2) The dye liquor is assumed as a Newtonian fluid, the viscosity is constant.
3) The influence of external volume force field, such as gravity is ignored.

The dye-liquor is assumed to circulate through the package. The direction of the flow is reversible either from out-to-in or in-to-out (Figure 1). Due to the symmetry of the package, the geometry of the system can be simplified to that shown in Figure 2. The symmetric line was treated as the left boundary of the system.

![Figure 2: Geometry of the system](image)

3.2 Parameters

The common parameter values used in the model are given below:
Package:
Inner radius 0.018m
Outer radius 0.056m
Height 0.136m
Others:
Fluid viscosity [6] 7.98e-4 kgm⁻¹s⁻¹
Package permeability [1] 3.8e-11 m²
Fluid density [6] 996 kgm⁻³

3.3 Solutions of the model based on Navier-Stokes/Darcy’s law

Equations 1, 3, and 4, combined with proper boundary conditions which are defined by assumptions and system geometry, were solved. The results are mainly presented in graphical form.

Figure 3 shows the flow velocity in both tube and yarn assembly. The colour and the length of the white arrow denotes the magnitude of the velocity, the white arrows indicate the velocity direction. It can be seen that there is a velocity direction change across the interface between the tube and yarn assembly. The velocities here are defined for x and y-directions in a two-dimensional model. However, they are based on the assumptions used in Darcy’s law, which simply ignore the influence of the shear stress.

Figure 4 demonstrates the velocity distribution across the cross section of the tube and yarn assembly (See figure 3, line y = 0). The resulting graph shows that the velocity is not continuous across the interface between the tube and yarn assembly. This indicates that a considerable change in the velocity of flow suddenly occurs in a region comparatively close to the surface of the yarns.

Figure 3: The distribution of flow velocity in tube and yarn bobbin, where the porous media is described by Darcy’s law

Figure 4: The distribution of flow velocity in the cross section (see figure 4, line y = 0) of tube and yarn

3.4 Solutions of equations based on Navier-Stokes/Brinkman model

Equations 2, 3, and 4, combined with proper boundary conditions were solved using Brinkman’s approach for porous media.

Figure 5: The distribution of flow velocity in tube and yarn bobbin, the porous media is described by Brinkman’s equation
It can be seen from figure 5 that the velocity in both magnitude and direction has a similar distribution to that shown in figure 3.

Figure 6 shows that the velocity is almost continuous across the interface between the tube and yarn assembly, with a sharp decrease just before the surface of the porous media. Also, the velocity distribution in tube phase is fluctuated, indicating different flow features, which is probably closer to reality, compared to that shown in figure 4. However, the velocity distribution in yarn assembly phase is almost the same as specified in figure 4. This is an important result, indicating that using the conditions employed in the simulation, the shear stress has a very weak influence on flow properties within the yarn assembly. In addition it is shown that the use of Darcy’s law to define the flow of dyeing liquor across the yarn package is quite reasonable. This also indicates that the flow rate within the yarns is quite low, and since in practice package dyeing machine, the boundaries defined in this simulation do not strictly follow the no-slip condition, this can lead to even lower velocities in the porous media than the results of this simulation suggest. Therefore the flow can be treated as laminar flow. Figure 7 further demonstrates the validity of the Darcy’s law when different flow rates are applied within a selected range.

Figure 7: Comparison of N-S/Brinkman and N-S/Darcy approaches
In figure 7, the solid and dashed lines denote the flow velocity at a flow rate of 0.1 m/s and 0.2 m/s respectively using N-S/Darcy’s approach; while the dot-dashed line and dotted line denote the flow velocity at a flow rate of 0.1 m/s and 0.2 m/s using N-S/Brinkman’s approach, respectively. This figure shows that the shape of the dyeing liquor flow within the tube for N-S/Brinkman approach is fluctuated, while that when N-S/Darcy’s method is employed is quite stable for both flow rates. The flow velocity within the yarn assembly, however, in both inflow rates, is almost the same using both N-S/Brinkman and N-S/Darcy’s approach.

In practice, we can increase the flow velocity within the yarn assembly by increasing the inflow rate. However, a higher flow rate will lead to an uneven distribution of flow velocity through the yarn assembly. Also, a higher flow rate causes a more fluctuated flow in the tube, which subsequently increases the influence of shear stress on flow properties, which is probably one of the reasons explaining the initial strike.

4. CONCLUSIONS

The Brinkman equations extend Darcy’s law to include a term that accounts for the viscous transport in the momentum balance and introduces the velocities in the spatial directions as dependent variables. This approach is more robust than Darcy’s law since it is valid in a bigger range of flow rates, and permeability of the porous media.

When investigating the dyeing liquor flowing through the package, since the flow rate is not high, and the permeability of the package is so low, the momentum transport by shear stresses can be ignored, thus the Darcy’s law is a reasonable approach to model the flow properties. This result is well known in this field, but it is the first time that is has been demonstrated on the basis of a sound mathematical model.

The higher the inflow rate provided, the higher the velocity in yarn assembly, and therefore a less even distribution of the flow velocity across the package will be obtained. If the evenly distributed flow velocity is important for a level dyeing, a high flow rate is not necessary in practice.

A high flow rate will cause a more fluctuated flow in tube, which increases the influence of shear stresses on flow properties. This is probably one of the reasons that could explain the initial strike.

In this simulation the flow velocity in the yarn assembly is only about 10 percent of that in inflow rate. In package dyeing machinery, an even lower velocity in porous media is expected due to the boundary condition. In real apparatus, strictly speaking, no-slip conditions do not apply to some boundaries. The flow therefore can be treated as laminar flow, and the dye dispersion through the package by diffusion cannot be neglected.

These flow models can also be used to design package dyeing apparatus, and select suitable material for the construction of tubes taking into account the static pressure predictions under different flow rates. The selection of correct type of flow metre, and pressure sensors based on their simulated range can also be carried out.

REFERENCES:


Author biography:
Dr. Shamey is currently a faculty member of the polymer and colour chemistry program at North Carolina State University. He is also a research supervisor of several PhD projects at Heriot-Watt University in UK. He has published several papers on this subject and is the founder of Automation and Novel Coloration research group at HW University. Mr Xiaoming Zhao is a member of the ANC group at HW University.
DIFFUSION OF DISPERSE DYSES INTO SUPERMICROFIBRES

K. H. Park and V. Koncar*
GEMTEX/ENSAT,
9, rue de l'Ermitage - BP 30329 - 59056 ROUBAIX CEDEX 01 FRANCE
E-mail : keun-hoo.park@ensait.fr, vladan.koncar@ensait.fr

KEYWORDS
Dyeing, diffusion, supermicrofibres

ABSTRACT

A common goal of all dyers is to achieve the correct shade as early as the first dyeing. But, the dyeing process is very complex, being characterised by the diffusion-controlled sorption of dyes that depends on several physicochemical parameters.

Moreover, the dyeing properties relating to microfibres and conventional fibres are caused by different properties such as exposed surface area and crystallinity index.

Wilson and Hill developed equations describing the uptake rate of disperse dyes by cylindrical fibre model as a function of the diffusion coefficient and the nature of the dyebath. The inverse of Shibusawa’s polynomial approximation of Hill’s and Wilson’s equation is used to compute the diffusion coefficient which depends on the initial dye concentration, the time and the fibre count for a fixed temperature.

In this paper, the sorption isotherms, the diffusion coefficient, the dye concentration evolution into the fibres and the dyeing uptake rate are computed from experimental results for conventional fibres, microfibres and supermicrofibres. The sorption isotherms and the diffusion coefficient evolutions as the function of time, initial dye concentration and temperature for supermicrofibres are discussed by considering the surface area and the diffusional boundary layer influence. Important differences of dyeing properties are observed, depending on the fibre count, which should be useful in the supermicrofibres dyeing process optimization. The aim is to provide a tool for dyeing practitioners to increase dyeing reproducibility and to improve the performance of “right-first-time” production.

INTRODUCTION

A microfibre is traditionally defined as a fibre or filament of linear density below approximately 1 dtex. However, even finer fibres of less than 0.3 dtex are produced, these commonly being referred to as supermicrofibres or ultrafine fibres (McIntyre and Daniels 1995).

It is obvious that the physical, mechanical and specially dyeing properties of these different microfibres 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 (Burkinshaw 1995; Shin and Bide 2000).

Moreover, the microfibres production method changes in function of the filament count. For instance, microfibres are often produced by direct spinning methods and supermicrofibres are rather produced by the conjugate spinning method.

Therefore, even if there is no difference in the chemical structure or morphology different microfibres exhibit several important differences related to dyeing properties. These differences are induced by different surface areas which has twofold influence.

First, the diffusion rate of disperse dye molecules into the fibre is different during the dyeing and secondly, the visual and instrumental depth of dyed microfibres are also different because of modified light reflection properties depending on the microfibre counts.

Most theoretical equations describing the overall rate of dyeing have been derived by assuming that this one is determined by the dye diffusion rate within a fibre (Cleve and Bach 1997; Crank, 1948; Etters and Urbanik 1983; Etters 1994; Shibusawa 1985; Wilson 1948).

Diffusion of dyes into polyester fibres can occur under both infinite and finite dyebath conditions during the dyeing process. In the case of infinite dyebath, the dye concentration in the bath does not change during the sorption process. In the case of finite dyebath, the dye concentration at the fibre surface continuously decreases during the sorption process until equilibrium between the dye concentration into polyester fibres and in the bath is achieved (McDowell and Weingarten 1969).

The diffusion rate of dye molecules into polyester fibres from a well-stirred solution is expressed by Wilson’s equation for a finite dyebath (1). Equation (1) is reduced to Hill’s equation (2) when fractional exhaustion \( E_{\text{fin}} = 0 \), i.e. describing the dyeing rate from an infinite dyebath.

\[
\frac{M}{M_\infty} = 1 - \sum_{n=1}^{\infty} \frac{4\alpha(1+\alpha)\exp(-q_n^2(D_f t/r^2))}{4 + 4\alpha + \alpha^2 q_n^2}, \quad (1)
\]
\[
\frac{M_t}{M_w} = 1 - \sum_{n=1}^{4} \frac{4}{\beta_n^2} \exp(-\beta_n^2 (D_e t / r^2)),
\]

where \(M_t\) and \(M_w\) are amounts of dye taken up by a fibre of radius \(r\) at time \(t\) and at equilibrium respectively, the \(\beta_n\) are positive (non-zero) roots of \(a_n J_1(\beta_n) + 2J_0(\beta_n) = 0\), where \(J_0\) and \(J_1\) are zero and first-order Bessel functions, \(D_e\) is the diffusion coefficient of dye in the fibre (cm²/s), the parameter \(a\) corresponds to \((1-E_v)/E_v\), where \(E_v\) is the fractional equilibrium exhaustion, and the value \(\beta_n\) are positive (non-zero) roots of \(J_0(\beta_n) = 0\).

The diffusion coefficient is computed from \(M_t/M_w\), experimental dyeing results for microfibres and conventional fibres. In the case of high initial dye concentrations, this infinite dyebath condition is maintained during all the dyeing process. In the case of lower initial dye concentrations, two different dyeing configurations are possible: transitional dyebath (first infinite and then finite dyebath) or only finite dyebath. As shown previously (Koncar and Casetta 2001; Casetta, Koncar and Cazé 2001), the dye diffusion into the fibre is more important during the finite dyebath enabling faster increase of \(M_t\) values. The fact is that the most important and efficient phase of a dyeing process is during the infinite dyebath condition. Therefore, in this paper, the diffusion coefficient is analyzed only for infinite dyebath configuration for supermicrofibres, microfibres and conventional fibres on the fixed dyeing temperatures (120 and 130°C).

**Experimental study**

**Fibres and dyeing procedure**

The polyester multifilament yarns, 0.07 dtex (Sea-Islands fibre type) 0.22 dtex (77 dtex/300 fil), 0.56 dtex (83 dtex/144 fil) and 1.46 dtex (292 dtex/200 fil) were used. All these fibres were obtained from Hyusung Inc., South Korea. The fibre radius \(R\) (cm) and the specific surface area \(A\) (cm²/g) were computed for all the fibres and are summarised in the Table 1.

<table>
<thead>
<tr>
<th>Parameter values</th>
<th>Parameter values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.07 dtex</td>
<td>0.22 dtex</td>
</tr>
<tr>
<td>Q, l/kg</td>
<td>50/1</td>
</tr>
<tr>
<td>R, cm</td>
<td>1.25 x 10⁻⁴</td>
</tr>
<tr>
<td>A, cm²/g</td>
<td>11.78 x 10⁻⁴</td>
</tr>
<tr>
<td>X, %</td>
<td>36</td>
</tr>
</tbody>
</table>

Dyeing of all polyester yarns was carried out with C.I. Disperse Blue 56 with further purification. The quantity of 14 g of yarn was used for the dyeing procedure in a well-stirred dyebath and the liquor ratio (Q) was 50:1. The dyings were carried out at two different temperatures 120 °C and 130 °C in order to fulfil the conditions of Hill’s equation. These temperatures were used in order to analyse the supermicrofibres dyeing properties in function of the dyebath temperature. In fact, initially, the yarn was placed in a dyebath at a temperature of about 90 °C. This was, then, heated up as quickly as possible (ca. 4 min) to 120 °C or 130 °C and maintained at this temperature for several hours (a time of dyeing). During this isothermal dyeing, dyebath samples were obtained regularly using the dyeing machine sampling device and any particulate dye still present in these samples was dissolved with dimethyl formamide. The absorbance of the dye solution samples was determined using a Datascan SF 600 PLUS spectrophotometer in order to evaluate the dye concentration in the dyebath in real time (applying the Beer-Lambert law). Then, from these results, the dye concentration within the fibre was computed. The initial dye concentrations range from 1 % (on the weight of fibre, o.w.f) to 16 %.

**Crystallinity index measurements**

The crystallinity index \(X\) was determined by differential scanning calorimetry (DSC 2920CE) measurements. Samples were examined with the temperature rate of 10 °C/min until 300 °C. After that, samples were put in the ice and the measurement procedure was repeated. For the crystallinity index the DSC curve was used to obtain the enthalpy values for the melting point and this value was divided by the theoretical enthalpy value for 100% crystalline polyester (Sichina 2000; Warner 1995).

**Sorption isotherms**

Sorption isotherms of disperse dyes, depending of various parameters (presence of dispersing agents, the dyeing temperature, the dye structure and fibre types) are reported as Nernst-type linear distribution isotherms in the presence of a constant concentration of dispersing agent (Schuler and Remington 1954).

However, in the real dyeing conditions, the sorption isotherms of disperse dyes on polyester fibres have more complex shapes. It has been reported that a dual mode sorption model (equation 3) is composed of both Nernst and Langmuir sorption models, may be more appropriate (Park, Cassetta and Koncar 2002; Shibusawa and Nakamura 1995; Nakamura, Rama and Kamiishi 2000; Shibusawa 1996).

\[
C_f = C_R + \sum_{n=1}^{n} C_{L_n} = K_R C_f + \sum_{n=1}^{n} \frac{K_{R_n} S C_0}{1 + K_{L_n} C_0} ,
\]

where \(C_f\) (g/kg) and \(C_R\), \(C_{L_n}\) are the equilibrium dye concentrations on the fibre and in the dyebath, \(C_R\) (g/kg) and \(C_{L_n}\) (g/kg) are the equilibrium concentrations of dye on the fibre sorbed by Nernst type partitioning and Langmuir sorption, respectively. \(S\) (g/kg) is the saturation value for Langmuir sorption, \(K_R\) (L/kg) is the partition coefficient and \(K_{L_n}\) (L/g) is the Langmuir constant. This model can be simplified to the simple dual-mode sorption model where \(n=1\).
In this study, the disperse dye isotherms and their sorption models were determined for supermicrofibres, microfibres and conventional polyester fibres (0.07, 0.22, 0.56 and 1.46 dtex) at two different dyeing temperatures (120 and 130 °C) in order to understand parameters evolutions.

**Dye uptake rate and diffusion coefficient**

In Hill’s and Wilson’s equations the diffusion coefficient is considered as constant. In fact the mean value of \(D_{i}\) (diffusion coefficient) have been used during the whole dyeing process. We consider the value of \(D_{i}\) as constant but only for a short period of time, when the temperature and the dye concentration variations are small. Therefore, inverse Hill’s equation is then used this short period to compute diffusion coefficient considered as constant. In fact, Shibusawa’s approximation of Hill’s equation has been utilised to compute the \(D_{i}\) values for the infinite dyebath phase. This procedure was detailed in our previous work (Park, Cassetta and Koncar 2002).

In this article, we have computed the diffusion coefficient evolution for two different dyebath temperatures and for different fibres counts.

Then, the obtained results were compared and analysed together with the corresponding dye uptake rate. Important differences were observed depending on the dyebath temperature, initial dye concentration and the fibre count. Moreover, several interesting conclusions confirming our previous studies have been pointed out.

The experimental parameters for all experiences realised during our research work are summarised in the Table 1.

**RESULTS AND DISCUSSIONS**

In this section, we give all the results concerning the sorption isotherms, dye uptake and diffusion coefficient differences with the discussions and an attempt to understand the phenomena that influence the dyeing process.

**Sorption isotherms for different fibres**

It has been shown that the sorption isotherms of fibres with counts of 0.22 dtex, 0.56 dtex and 1.46 dtex at a dyebath temperature of 130 °C are mostly double Langmuir type (Shibusawa 1996).

Many results in the literature show that the contribution of the Langmuir sorption isotherm to the dual mode sorption model increases with an increase in dyebath temperature and a decrease the dye molecules (Nakamura, Ohwaki and Shibusawa 1995; Etters 1991; Shibusawa 1980).

The sorption isotherms for this present study for the super microfibre (0.07 dtex) at two different dyeing temperatures (120 and 130 °C) are shown in Figure 1.

All sorption parameters are given in the Table 2 for 120 °C and in the Table 3 for 130 °C, respectively, where \(R (%)\) is the degree of correlation between experimental results and sorption isotherm model.

For the super microfibre, the sorption isotherm fit to the simple dual mode sorption model composed of the Nernst and the first order \((n=1)\) Langmuir sorption models.

It is interesting to point out that for the super microfibre (0.07 dtex) the parameters \(K_p, K_L, l\) and \(S\) decrease as the dyebath temperature decreases.

Also, the only Langmuir sorption based model is not sufficient to describe the supermicrofibres sorption isotherms.

Table 2: Sorption parameters for 0.07 dtex supermicrofibre, 0.22 dtex and 0.56 dtex microfibres and 1.46 dtex conventional fibres at 120 °C

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Sorption parameter values at 130°C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.07 dtex</td>
</tr>
<tr>
<td>(K_p), L/g</td>
<td>153</td>
</tr>
<tr>
<td>(K_L), L/g</td>
<td>14.11</td>
</tr>
<tr>
<td>(S_i), g/kg</td>
<td>83.87</td>
</tr>
<tr>
<td>(K_L), L/g</td>
<td>-</td>
</tr>
<tr>
<td>(S_i), g/kg</td>
<td>-</td>
</tr>
<tr>
<td>(R) %</td>
<td>99.89</td>
</tr>
</tbody>
</table>

Table 3: Sorption parameters for 0.07 dtex supermicrofibre, 0.22 dtex and 0.56 dtex microfibres and 1.46 dtex conventional fibres at 130 °C

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Sorption parameter values at 120°C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.07 dtex</td>
</tr>
<tr>
<td>(K_p), L/g</td>
<td>90.23</td>
</tr>
<tr>
<td>(K_L), L/g</td>
<td>8.60</td>
</tr>
<tr>
<td>(S_i), g/kg</td>
<td>80.23</td>
</tr>
<tr>
<td>(K_L), L/g</td>
<td>-</td>
</tr>
<tr>
<td>(S_i), g/kg</td>
<td>-</td>
</tr>
<tr>
<td>(R) %</td>
<td>99.63</td>
</tr>
</tbody>
</table>

Figure 1: Sorption isotherms for super microfibre (0.07 dtex) at two different dyeing temperatures (120 and 130 °C)
In this case, the sorption isotherms have to be modelled by a simple dual mode Nernst and Langmuir sorption models for 120 and 130 °C.

Dye uptake and diffusion coefficient analysis

The rate of dye uptake \((M_t/M_{\infty})\) and the sorption value of \(M_t\) \((M_{\infty} = \lim_{t \to \infty} M_t)\) were investigated for different fibres, different initial dye concentrations and at two dyebath temperatures (120 and 130°C). These differences were analysed and discussed in this section. After that, the diffusion coefficients for all the cases were computed and compared. In fact, the diffusion coefficient evolution, obtained from the inverse Hill’s model corresponding to the infinite dyebath phase, describes the dye uptake rate evolution in the first phase of dyeing process.

The dye uptake rate evolutions for all the fibres (0.07 dtex, 0.22 dtex, 0.56 dtex and 1.46 dtex) at 120 °C dyebath temperature and for the initial dye concentrations of 8 % o.w.f. are compared in Figure 2. The same structure of dye uptake rates at 130 °C is shown in Figure 3.

Figure 2: Dye amount evolution \((M_t)\) vs. time for different fibres for 8 % initial dye concentration at 120 °C

It is clear that the dye uptake rate at time \(t\), before the equilibrium, \((M_t)\) increases faster as the fibre count decreases. This phenomenon is valid for both dyebath temperature and it can be explained by the larger surface area exposed to dye molecules for supermicrofibres than that for microfibres or conventional fibres. However, this rate phenomenon is more important at the lower temperature 120 °C and this can be explained by the contribution of surface area which is more important at 120 °C. At 130 °C the dyebath temperature contribution importance becomes more significant. Therefore, as the diffusion of dye molecules is more important at 130 °C, the surface area exposed to dye molecules has less importance and differences of dye uptake rate evolutions \((M_t)\) are smaller.

Another important parameter of supermicrofibres dyeing process is the saturation value of \(M_t\) (i.e. \(M_{\infty}\)).

From Figures 2 and 3, it is clear that the \(M_{\infty}\) value increases as the dyebath temperature increase and especially as the polyester fibre count decreases.

Thus, the statement that the supermicrofibres have higher dyeing capacity than microfibres (and microfibres have higher dyeing capacity than conventional fibres) due to the quantity of the available dyeing sites is confirmed. For higher initial dye concentrations and higher dyebath temperatures, there are more molecules to be fixed on available dyeing sites.

The diffusion coefficient \((D_t)\) for the supermicrofibres, microfibres and conventional fibres were analysed only during the infinite dyebath phase for infinite and transitional dyebaths. The diffusion coefficient for the finite dyebath configuration, corresponding to very low initial dye concentrations due to their low values, has not been studied. In fact, the diffusion coefficient for infinite dyebath configuration illustrates the slope of dye uptake rate during the initial stage. Therefore, the diffusion coefficient measures the intensity of variations of \(M_t/M_{\infty}\) and not the variations of dye quantity into the polyester fibres or microfibres. Thus, the diffusion coefficient is relative parameter.

Consequently, a dyeing process is completely defined by the diffusion coefficient \(D_t\) and the saturation dye uptake \(M_{\infty}\).

As in our previous work (Park and Koncar 2002), the first observation is that for transitional baths the duration of the infinite dyebath phase decreases as the fibre count decreases. It is specially marked for a supermicrofibre, where the dyeing capacity is extremely high and the amount of dye molecules penetrating the supermicrofibres is important in the beginning of dyeing.

Finally, in the Figure 4, the case of the supermicrofibre (0.07 dtex) with the 16 % o.w.f initial dye concentration (infinite dyebath) at two different dyebath temperature (120 °C and 130 °C) is exposed. It is evident that the
temperature of bath influences the diffusion coefficient. In fact, the diffusion is better at higher temperature. This is verified for all the fibre counts.

![Graph showing diffusion coefficient vs time for different temperatures](image)

Figure 4: $D_t$ for different dyebath temperatures (120 °C and 130 °C) at 16 % initial dye concentration for supermicrofibre.

CONCLUSIONS

The first conclusion of this study is that the sorption isotherms of supermicrofibres are different from those of microfibres and conventional fibres. This means that the dyeing process of supermicrofibres is different and that these kinds of fibres have to be dyed with precautions.

The second conclusion confirms the results of our previous study. Therefore, the surface area and different dyeing capacities of supermicrofibres can explain the changes in the dyeing uptake rates. It is obvious that supermicrofibres can accept more dyes and that they have to be dyed at higher temperatures.

REFERENCES


BIOGRAPHY

KEUN-HOO PARK was born in 1974. He obtained his Master’s degree in February 1999 at the University of Chonnam National in South Korea and his PhD in April 2004 at the University of Lille 1, France in department of Instrument and advanced analysis, in field of modelling and the simulation of dyeing process for polyester microfibres.

His research fields of interest are: Predictive control for textile processes, modelling of dyeing processes, nonlinear optimization and interactive fibres.

VLADAN KONCAR was born in 1962. He obtained his Master’s degree in June 1987 at the University of Lille 1, France, Automation Control Laboratory and his PhD in January 1991 at the University of Lille 1, France in the field of Automation Control - Multirate digital control and large scale discrete system control.

He moved in 1992 at the Textile Engineering Institute ENSAIT where he is now full time Professor.

His research fields of interest are: Predictive control, Compound control, Robust control of complex systems, modelling of dyeing processes, nonlinear optimization, Virtual reality and communicated objects.
FIBRE TEXTILE APPLICATIONS
NONLINEAR MODELLING OF FIBRE SPINNING
AIMED AT FASTER PRODUCTION AND PRODUCT DEVELOPMENT

Bulsari, Abhay
Nonlinear Solutions Oy, PL 953, 20101 Turku 10, Finland
abulsari@abo.fi

Prof. Harlin, Ali
Puolakka, Arja
Tampere University of Technology
PL 589, 33101 Tampere, Finland
ali.harlin@tut.fi

KEYWORDS
Industrial control, Industrial process, Performance analysis, Neural network, Nonlinear analysis

INTRODUCTION
Textile industry continuously strives to produce better products from better materials, and fibres with higher and higher demands on its properties from the available raw materials. Most of this kind of research is being done today by trial and error, resulting in extensive experimentation. This paper illustrates how from a limited number of systematically designed experiments, nonlinear models describe the quantitative effects of process variables of spinning on the fibre properties. In combination with a suitable software tool, it becomes relatively easy to determine the spinning process variables which result in desired fibre properties, whenever possible. The nonlinear models also serve to simulate experiments, thus reducing the time and costs associated with experimentation and testing. The article discusses nonlinear modeling in melt spinning of polypropylene fibre for technical and hygienic products. Main issue is to optimize product weight, tenacity, processability, and specific properties, which enable reduction of energy and raw material consumption.

POLYPROPYLENE MATERIAL
Polypropylene consumption in fibre materials is mainly in technical and hygienic textiles. The fibre is manufactured by means of melt spinning and blown of MFR2 = 15…50 and 350…800g/10min respectively. The fibre materials are light, strong, and hydroscopic. They are insect, chemical and solvent resistant. The crystallinity of polypropylene depends on fibre manufacturing like polymer melt temperature, cooling rate, and tension in fibers during the spinning process (Ahmed 1982, Warner 1995).

MELT SPINNING
In the melt spinning the extruder melts, coeys, and homogenize polymer. Extruder barrel temperature is controlled in zone, but the actual melting and mixing energy is distributed through extruder screw. Melt-pump pressurize and stabilize flow through the spinnerets. Spinning is compact or long according to cooling length of <100mm and >1000mm respectively. (Edana 2000)

Mechanical properties
Mechanical properties of polypropylene depend on the amount of crystallinity and crystal size. The material hardness, strength, and tensile at yield are increased, when the crystallinity is increased. Increase in polymer molar mass increase material toughness, melt strength, and maximum elongation, but reduces the density through more complicated crystallization under rapid cooling conditions. Orientation has marked effect on tensile properties.

Fibre orientation
Molecule orientation affects on both polymer physical and optical properties. On the other hand orientation influence also in polymer crystallization, solidification kinetics, and deformation dynamics. In the spinning process orientation is an object of drawn speed induced deformation (crystal orientation) and temperature based relaxation (heat movement). Main increasing factor for the crystallization is forced molecular orientation. (Gugumus 1993)

The final orientation is result of extensional flow and streaming orientation, which reach its kinetically stable form under competitive kinetics of crystallization and relaxation. Spinnert has minor or no influence on the total orientation (Billmayer 1971, Lieberman 1996).

Cooling rate is controlled by means of cooling air rate and temperature, which affects the crystallinity through crystallization time. The higher is the crystallinity the
less is elongation even if fibre strength should remain practically the same (Edshammer 1995).

Fibre physical properties can be influenced especially where the crystallization starts, typically at low spinning speeds (Turbak 1993). Spinning tension defines the amount of amorphous and crystalline and thus also the mechanical properties. The tension is result of the first galet speed, polymer molar mass, extrusion temperature, productivity, as well cooling air speed and temperature. Increase in molar mass leads increase in spinning tension, but also in crystallization rate, crystallinity, orientation, and fibre strength while elongation is reduced (Lorando 2002).

It has been also observed that the first galet speed, extrusion temperature, and polymer melt index all affect on where the crystallization front starts. The temperature for crystallization onset is increased while the first galet speed, polymer molar mass, or even extrusion temperature is increased. In the spinning process the average drawn speed E is defined as follows:

\[ E = \frac{(V_g - V_c)}{L_m} \]

Where
- \( V_g \) = first galet speed
- \( V_c \) = melt speed straight after spinneret
- \( L_m \) = deform length over the spin line

Spinning tension is directly dependant on E and average drawn viscosity \( \chi \). While either E or \( \chi \) are increasing, the tension is increased accordingly resulting also higher fibre orientation. Under Hagler model the following dependencies are valid /16/.

- The first galet speed increases; E increases
  \( \Rightarrow \) Orientation increases
- Molar mass increases; \( \chi \) increases
  \( \Rightarrow \) Orientation increases
- Spinning temperature decreases; \( \chi \) increases
  \( \Rightarrow \) Orientation increases
- Air flow rate increases; \( L_m \) decreases
  \( \Rightarrow \) Both E and orientation increase
- Spinneret hole size increases; \( V_c \) decreases
  \( \Rightarrow \) Orientation increases

**Post orientation**

Fibre bundles are post oriented with a pair of rollers, the later of which has greater spinning speed. Primary orientation is molecular organization in direction of the drawn, which result the polymer primary bonds oriented in fibre length vice and only the secondary bonding cross the fibre. Crystalline polymers are elongated below their melting temperature, in case of polypropylene typically 600-800%.

Under elongation crystal lamellas break down in smaller crystal domains and some of the amorphous material may crystallize. Heating the fibre increases the crystal size. Crystallinity may increase or decrease according to the elongation condition, however in elevated temperature typically crystallinity is increased.

**EXPERIMENTAL**

Experiments were made with Fourne pilot fibre spinning line. The main extruder size is 16mm, and the screw length L/D= 20. The line is equipped with melt pump and eight hole spinneret with holes of size 0,3mm size. The cooling section length in the cooling chamber was 1800mm. Spinneret take-up speed was adjustable between 100 and 500rpm. In the system is one additive feeding unit for oil and lanolin acid feed.

Because the extruder size is small and the effect of external heating is marked, the polymer melt temperature was adjusted by the extruder melt temperature. Melt pump was adjusted to constant speed, and the extruder screw speed was controlled to provide 45 bar in pump inlet. Line control was able to adjust the pressure on spinneret, except at the lowest pump speed changed from original 10 rpm up to final 12 rpm. Cooling air was of temperature 21oC and humidity 40%. The cooling airflow was adjusted according the cross sectional area in the air distributor in cooling chamber. Draw down ratio was adjusted with spinning take up galet 1 and 2 speed difference. The galet temperature 70 oC was utilized.

A set of experiments was performed in order to model fibre quality according to the spinning parameters. One base polymer material was utilized. The measured fibre quality outcome parameters were fibre titer, tenacity, elongation, and force at yield. The experiment plan is based on four variables, namely

- **Melt temperature**, 260…290 oC
- **Melt pump speed**, 10…25 rpm
- **Cooling air in chamber**, 0…100%
- **Draw down ratio**, 4…10

Borealis manufactured the polymer material used in the experiments. The actual grade name was HG245FB, which is rheology adjusted Ziegler-Natta type homopolymer has the melt flow rate of MFR 2,16; 230 = 26 g/10min. The material is designed for fine filament count stable fibre and continuous filament. The experiments were mainly for MOY and FOY fibre.
The raw data revealed a clear dependence between the titer and pump speed, and draw down ratio. Tenacity is increased and elongation is decreased as a function on draw down ratio. The effects were expected. Melt temperature had slight logic negative effect on titer and elongation, but airflow seem to have minor or no effect at all. The results differentiate from the one in short fibre spinning but can be accepted for slow speed and long cooling fibre spinning.

Fluctuation in experimental results due to measurement accuracy or environmental disturbance was better than 10% for all out comes except for tenacity better than 5%. Typical values were 7...9% and 3...4% respectively. Major difficulty was caused by the processing additive, which was first slipping in the tensile measurement finally washed with alcohol away from the fibre. The fibre titer was measured by ultrasonic Vibroshop device (Lenzig AG) and the tensile with Vibrodyn device in the conditioned room after 48 hours of fibre production.

Effect on birefringence was non-monotonic and accurate model was difficult to conduct. However results DSC measured melting enthalpies are well correlating non-linearly on crystallization conditions of the drawn fibre. As shown in the figures 2 to 5 can be seen nearly threshold functions on enthalpy on respect of melt temperature, melt pump speed, cooling control level, and draw down. Enthalpy of used non-oriented polypropylene reference was 85,6 J/g. Fibre orientation can be seen in differentiation in the enthalpy related to crystallinity in the polymer material.
Between the melt pump speed and melt temperature is close correlation. Both draw ratio and cooling efficiency indicate clear threshold value for orientation. The pattern, which defines MOY and LOY fibre is markedly production line dependant and most complicated to be modeled through traditional methods.

**MATHEMATICAL MODELLING**

Mathematical modeling is performed in several ways. Physical modeling is suitable for certain situations. Empirical modeling is suitable for some situations. Many situations allow for combinations of these. While modeling fibre properties from process variables, the choice is limited. Physical modeling is too complicated for the purpose, and anyway involves too many assumptions and simplifications. Empirical modeling is therefore the viable alternative. Conventional linear modeling is clearly not wise for this problem since the effects of some process variables are clearly nonlinear. Consequently, nonlinear modeling has been used for the purpose.

**New techniques of nonlinear modeling**

New techniques of nonlinear modeling which have come up within the last ten-twelve years, have made possible the development of highly sophisticated nonlinear empirical models, without knowing the type and severity of nonlinearities present in the relations. It is also possible to combine process knowledge with this kind of empirical models, which often leads to better models. These new techniques have opened up new possibilities. It is now possible to develop accurate and reliable nonlinear models of material properties like tensile modulus and impact strength from experimental or production data, when the data has sufficient information content.

There are hardly any processes or materials in this world, which have absolutely linear characteristics. It is therefore wise to treat the nonlinearities rather than ignore them. To treat the nonlinearities, one can use new techniques of nonlinear modeling, like artificial neural networks. The proponents of linear techniques draw on their simplicity and the possibility of adding nonlinear terms in linear regression. Often this is not done, and is not efficient even if it is done. Nature does not follow the simplicities that we try to fit it in, using linear techniques.

Neural networks, on the other hand, have the so-called universal approximation (Hornik 1989) capability, which make them suitable for most function approximation tasks we come across in process industries. The user does not need to know the type and severity of nonlinearities while developing the models.

**Nonlinear modeling and neural networks**

Nonlinear modeling can be roughly defined as empirical modeling which takes at least some nonlinearities into account. Nonlinear modeling can be performed in many ways. The simpler ways include polynomial regression and linear regression with nonlinear terms. One can also use basis functions and splines, and in cases where the form of the nonlinearities is known, nonlinear regression can be used. Artificial neural networks are a set of efficient tools for nonlinear modeling, for reasons mentioned earlier, particularly the universal approximation capability of feed-forward neural networks.

There are many different types of neural networks, and some of them have practical uses in process industries (Bulsari 1995, 1998). Neural networks have been in use in process industries for about ten years. The multi-layer perceptron is a kind of a feed-forward neural network. Most neural network applications in industries (Bulsari 1999, 2000, Hooli 2000, Käppi 1998) are based on them. They have been used for nonlinear modeling of material properties of steels, copper, plastics, concrete, paper, pulp, etc.

Nonlinear modeling has its disadvantages too. Nonlinear models of good quality are not developed in a week or a month. It takes several months to develop highly reliable and robust nonlinear models, and needs special expertise, which is rare. A lot of people claim to be able to develop neural network models and can offer you nice looking user interfaces which don’t show the details of the models. However, not many can offer models of a high quality. This makes nonlinear modeling quite expensive. Still, as evinced by several cases, the benefits clearly outweigh the expenses.

Nonlinear modeling can contribute a lot in polymer processing (Bulsari 2002). Nonlinear models can help improve the properties of products produced with almost all the polymer engineering processes including injection moulding, extrusion, melt spinning, wet spinning and blow moulding. Nonlinear modelling can also help in designing polymer blend compositions and composites to produce the desired properties.

Artificial neural networks resemble structurally and to a smaller extent functionally the networks of neurons in biological systems. Like the networks of neurons in the brains, artificial neural networks also consist of neurons in layers directionally connected to others in the adjacent layers (see Figure 6).
It is also possible to combine neural networks with physical models or other empirical models, which often lead to better solutions.

**Modelling results**

Several runs were attempted with NLS 031 nonlinear modelling software with different configurations for the models. Since the experimental data was of good quality, and contained the necessary information for determining the nonlinearities, it was relatively easy to develop models for each of the four properties of spun fibres. Some aspects of the results are described in the sequel. The correlation (fraction of the variance explained by the model) was large for all the four properties (see Figure 7), and the rms % errors were comparable to the measurement accuracies. Figure 8 shows a plot of predicted vs measured values of tenacity. Figure 9 shows a histogram of prediction errors for one nonlinear model of elongation. It is good to see that even from this relatively small number of observations, the distribution is close to Gaussian. Nonlinear models are expected to be more accurate than linear models. Figure 10 confirms that one simple nonlinear model is much more accurate than the linear model for force at break.

![Figure 7. All the four models have a high degree of correlation (scale on the right)](image)

The behaviour of the nonlinear models seems reasonable. Draw ratio shows the most significant effect for all the four properties. For titer, pump speed has an equally significant effect. Increasing the draw ratio increases the orientation of the fibres, which increases tenacity and decreases elongation at break. These effects are visible from Figures 11 and 12, which are plotted keeping other variables constant.

Crystallinity models have increased level of error compared to the models of the mechanical models, which may due to characteristics of sampling for DSC and on the otherhand process limiting factors at line limiting speeds. However the results explain well the observations and are consistent with the idea of controlled crystallinity through draw ratio. See figures 13 and 14.
**LUMET systems**

Advanced and complicated models like the ones often implemented in LUMET systems can be cumbersome to use in their raw form. Unlike simple regression models, the equations look clumsy and the free parameters don’t tell you very much unless you are very familiar with these kinds of nonlinear models. LUMET systems make it easy for the user to utilise the models without needing to understand the technology in detail. At the same time, these are no black boxes.

The second version of the LUMET systems can implement, in principle, almost any kind of empirical models, but they are particularly designed for nonlinear models based on neural networks. The models might also be combinations of neural networks with other kinds of models. Plain linear regression models can also be implemented.

**Figure 8.** Measured values of tenacity match well with those predicted by one nonlinear model.

**Figure 9.** A histogram of prediction errors of elongation at break

If the user wants to see what the models are doing, a few clicks of the mouse will also show what is going on inside. Over the years, we have understood better and better the facilities, which make the system more valuable for the user, and therefore, the LUMET systems of today are a result of almost continuous improvement of its features.

**Figure 10.** Nonlinear model for force at break outperforms the linear model in terms of accuracy.

**Figure 11.** Effect of draw down ratio on tenacity as predicted in the LUMET system

**Figure 12.** Effect of draw ratio on elongation at break as predicted in the LUMET system

**Figure 13.** Effect of pump speed on DSC enthalpy as predicted in the LUMET system
CONCLUSIONS

Titer can be explained according to melt pump speed and draw down ratio. The higher is the productivity the thicker is the fibre, and other way the more it is drawn the thinner it becomes. Tenacity and elongation both depend mainly on the draw down, while it defines the crystal orientation in the fibre. Finally the force at fibre yield is markedly non-linear and depending on pump speed and draw down ratio. All the results are reasonable and understood based on theory.

With the developed model predict the fibre quality expected by means of tenacity and elongation and the respective operation conditions. Nonlinear modeling provides an effective tool to control various kinds of processes including fibre spinning. The accuracy in the limits of experimental values is available, however further the point accuracy is lost. Reasons as differentiation in heat transfer, crystallization and orientation mechanism are drastically changed when used ultimate, but not desired operation conditions.

References

Bulsari A. 1995, Neural Networks for Chemical Engineers, Elsevier.
Bulsari, A. et al., 2002 "Nonlinear modelling paves the way to bespoke polymers", British Plastics and Rubber, December, 4-5.
EDANA, 2000, Nonwovens Training Course.


Biographies

Prof. Ali Harlin, D.Sc. has born 1963. He received master of Science degree in Chemical Engineering (Polymer Science) in 1987 and the degree of Dr. Tech. in Polymer Technology (Extrusion technology) in 1996 at Helsinki University of Technology, in Finland. Before joining Nextrom, he worked in Borealis (former NESTE Chemicals) as a project manager and development manager. Since 1997 he was with Nextrom as General Manager, Technology & Development. Finally 2000 he was nominated Managing Director of Conex Cables Oy. Prfessor Harlin is responsible of technical textiles in the Fibre Material Science of Tampere University of Technology. He is also docent of high voltage engineering (materials) in Helsinki University of Technology.

Dr. A. Bulsari has a doctorate in chemical engineering from the University of Virginia (1988). He has edited 6 books, several special issues of journals, chaired the EANN series of conferences until 1998, and written over a hundred papers. Since 1996, he has provided numerous solutions to process industries, mostly based on nonlinear modelling. Polymers has been the largest industrial sector in his activities in the last few years.

Arja Puolakka has received Master of Science (Tech.) degree in Machine Engineering (Textile and Clothing) in 1982 at Tampere University of Technology in Finland. She has worked at Tampere University of Technology as an assistant of chemical textile technology and research scientist 1984-1986 and 1988-. In 1987-1988 she has worked as a laboratory engineeer in Tamfelt.
ELECTRO-MECHANICAL AND MORPHOLOGICAL PROPERTIES OF PET CONDUCTIVE POLYMER FIBRES – MODELLING AND SIMULATION

Bohwon Kim, Vladan Koncar and Pascal Bruniaux
Laboratoire de Génie et Matériaux Textiles (GEMTEX),
Ecole Nationale Supérieure des Arts et Industries Textiles (ENSAIT),
9 rue de l’Ermitage BP 30329, 59056 Roubaix, France
kim.bohwon@ensait.fr, vladan.koncar@ensait.fr, pascal.bruniaux@ensait.fr

KEYWORDS
Conductive fibre, Modeling, Polyaniline (PANI), Coating, Stretching, Stress-strain curve.

ABSTRACT

Conductive fibres were obtained using chemical coating based on polyaniline (PANI) and polyethylene terephthalate (PET) fibres. The electrical conductivity morphological characteristics and traction behaviour of conductive yarns obtained were investigated. Moreover, the mathematical models of electrical and mechanical properties have been developed. These conductive yarns are supposed to be used to realize connections in smart clothing or conductive fabrics for electromagnetic shielding applications.

INTRODUCTION

Recently, the textile industry has made considerable advances in the field of high value added textiles, mainly in the sectors of high performance textiles and yarns. The term “intelligent apparel” describes a class of apparel that has active functions in addition to traditional properties of clothing. These novel functions or properties are obtained by utilizing special textiles or electronic devices integrated into the textile structures. Many intelligent textiles already feature in advanced types of clothing, principally for protection and safety and for added fashion or convenience. The broad principle topics can be simply noted; professionals (the need for “free hands function”), health care (monitoring, training result diagrams), every life (telephone, wellness), sports (training, performances measurement) and leisure (aesthetic customization network games) (Bauer 1999; Science 2003). Moreover, the demand for electrically conductive fibres and textiles structures used for industrial materials is growing up in the domains such as sensors, electrostatic discharge, welding of plastics, electromagnetic interference shielding, dust and germ-free clothing (Park and Chang 1996), data transfer in clothing but also for military applications like camouflage and stealth technology.

For all the applications previously mentioned the basic element is of course the textile fibre. Therefore, it is important to modify the fibres properties (mechanical and electrical) in order to modify general textile properties and to add previously mentioned new functions. Modification of fibres based on conductive polymers seems to be the interesting approach enabling these new functionalities. In other hand, conductive polymers exhibit interesting electrical properties and wide variation in colour due to their conjugated double bonds chain structures, which derive from both their conducting or neutral (non-conducting) forms. However, they are inherently insoluble and infusible due to their strong intermolecular interactions. High quality conducting blends with conventional polymers by melt mixing (Zilberman et al. 1998; Hosier et al. 2001; Yang et al. 1998) or by solution casting (Yang et al. 1993) are still in a development stage.

Among the conjugated conducting polymers, polyaniline (PANI) has attracted much interest worldwide because of its excellent environmental, thermal and chemical stability. In addition, with simple doping process by charge transfer or by the acid/base (protonation) reaction, polyaniline converts the semi conducting emeraldine base form (10^{-5} S/cm) to the metallic emeraldine salt form (10^{-2} S/cm). In this paper a doped polyaniline has been used in order to create conductive fibres with textile properties.

The main objective of our research work was to obtain improved electrical conductive fibres (yarns) without modifying their mechanical (textile) properties. These textile properties are very important because the conductive fibres should be transformed in textile structure by weaving, knitting or other manufacturing process. This paper consists by two sections. In first part, the electrical and morphological properties of conductive yarns were studied. In second section, the basic theories of modelling as well as the results of simulation concerning our conductive yarns were described.

EXPERIMENTAL STUDY

Materials

Emeraldine salt form polyaniline (PANI-ES) was supplied by Sigma-Aldrich Chemical Company. Dodecylbenzene sulfonic acid (DBSA, Fulka) was used as the dopant to increase the conductivity and to obtain a homogeneous conducting solution of PANI in xylene. In the sol-gel process, PET spun fibre (Hyosung) and multifilament PET yarn (Hyosung) were used. PET yarns were used after washing with the solution of 2g/l of sodium carbonate and 1g/l of Tinovetine at 70°C for 30 min.

Solution preparation and coating process
PANI-DBSA mixtures with 25 % PANI-ES were dissolved in xylene to prepare 3, 5, 6 and 10 wt % PANI-DBSA/xylene solutions. These solutions were stirred vigorously at 120°C for 3 hours. Then, they were treated in the ultrasonic bath at 80°C for 2 hours. PANI coating was carried out during the impregnation of PET yarns in the PANI xylene solutions. This process has been performed using an apparatus that can control the temperature and the taken-up speed. This process may contain several baths filled with PANI-DBSA/xylene solution. The temperature of these baths varies from 30 to 100°C. PET yarns were withdrawn through the baths in the presence of a dry airflow along the coating surfaces to enhance solvent evaporation. The taken-up speed of yarn is adapted to the drying dynamics and for our experiments; it was set to 6 cm/min.

**Characterization**

The electrical resistance tests were performed using the “two-probe test technique” with Keithley 617 electrometer used as an ammeter at 25°C, 55.56 HR %. Keithley 617 electrometer generates the variable voltage DC powers supply connected to the conductive fibres. In the same time, it measures the electric current with the pA maximal precision. The electrometer is connected to a PC computer, which controls the voltage of DC power supply and records the measured current. Detail images of the PANI coated PET yarn surfaces were obtained using SEM (JEOL 100 CX model ASID4-D Scanning Electron Microscope). The SEM samples were gold-sputtered before observation. Tensile strength test was performed using the MTS tensile test machine (with TestWorks 4 program) for 5 cm of conductive yarns at 100cm/min-1 displacement speed.

**RESULTS**

**SECTION 1: Electrical and morphological properties**

All the electrical resistances of PANI-PET yarns obtained by coating with PANI solutions are presented in the Table 1. The electrical resistance was calculated from intensity (I) vs. applied voltage (V) measurements when V varies from −5 V to 5 V. In all the curves I vs. V characteristics are linear. The experiments have been repeated 30 times for different fibre lengths from 1 cm to 50 cm and average values have been computed.

![Figure 1: SEM micrographs of 10 % PANI solution coated PET (a) spun fibre yarn and (b) filament yarn (× 2000)](image)

From Table 1, it is possible to observe that the electrical conductivity shows two different areas delimited by the PANI solution concentration. The electrical resistance (Ω) below 5% PANI solution coated PET fibres is high and the conductivity of 6% PANI solution coated PET becomes significantly better. The electrical resistance of PANI/spun 10% has lower values than that of PANI/spun 6%. The electrical property for filaments yarns shows similar result with spun fibre yarn. We can notice that the high concentrations of PANI solution are very important for the conductivity increase of conductive yarn. We can observe the PANI aggregation on the surface of spun and filament yarns in the Figure 2. Moreover, a darkening on the PET substrate was observed and this suggests that PANI is absorbed onto the fibre substrate during the absorption of PANI solution.

**SECTION 2: Modelling of conductive fibres electrical properties**

**Electrical model for PANI/PET spun fibre yarns**

It is obvious from Tables 1 that the electrical resistance of conductive spun fibre yarns depends on the distance between electrodes linearly for all the PANI concentrations. The Figure 2 is composed of 5 graphics showing the electrical resistance R of PANI spun fibre yarns in function of the PANI concentration P and for different distances d between two electrodes of Keithley

<table>
<thead>
<tr>
<th>Resistance (Ω)</th>
<th>PANI/PEI spun yarn length (cm)</th>
<th>Concentration of used PANI solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3%</td>
<td>5%</td>
</tr>
<tr>
<td>1 cm</td>
<td>1.70 x10^5</td>
<td>1.35 x10^5</td>
</tr>
<tr>
<td>3 cm</td>
<td>2.38 x10^5</td>
<td>3.79 x10^6</td>
</tr>
<tr>
<td>5 cm</td>
<td>5.29 x10^5</td>
<td>6.96 x10^6</td>
</tr>
<tr>
<td>10 cm</td>
<td>1.53 x10^6</td>
<td>1.96 x10^7</td>
</tr>
<tr>
<td>15 cm</td>
<td>2.54 x10^6</td>
<td>3.47 x10^9</td>
</tr>
<tr>
<td>20 cm</td>
<td>3.92 x10^6</td>
<td>4.65 x10^11</td>
</tr>
<tr>
<td>50 cm</td>
<td>9.95 x10^6</td>
<td>7.23 x10^13</td>
</tr>
</tbody>
</table>
617 electrometer (d is a fibre length). In the same graphics the simulated electrical resistance of fibres (in dashed line) is presented. The simulation is based on the mathematical model given below:

\[ R = R_0 \cdot P^{-\beta} \]
\[ R_0 = A \cdot \exp(\alpha \cdot d) \]

Where,
- \( d \): fibre length (cm),
- \( \alpha \): electrical conductivity coefficient (cm\(^{-1}\)),
- \( A \): PET spun fibre apparent resistance,
- \( P \): PANI solution concentration,
- \( \beta \): PANI correction coefficient,
- \( R \): fibre electrical resistance.

The identification of \( A, \alpha \) and \( \beta \) parameters based on the electrical resistance measured values for PANI spun fibre yarns leads to: \( A = 3 \times 10^8 \) \( \Omega \), \( \alpha = 0.162 \) cm\(^{-1}\) and \( \beta = -6.5 \).

**Electrical model PANI filament fibres**

In the case of PANI filament fibre yarns the mathematical model of electrical resistance has not been developed due to the lack of experimental data. It is however possible to observe from the Table 2 that the electrical conductivity shows two different states delimited by the PANI concentration. Below 6% the PANI filament fibre yarn conductivity is week (2.76x10\(^4\) \( \Omega/cm \)) and for 6% and more it becomes better (7.83x10\(^2\) \( \Omega/cm \). There is a threshold value of PANI concentration between 5% and 6% of PANI solution.

**Figure 2:** The electrical resistance (R) of PANI spun fibre yarns in function of the PANI solution concentration (P) and for different fibre distances (d) between two electrodes.

**Mechanical model for conductive fibres**

A dynamic behaviour study of these two new yarns has been carried out. The main goal is to verify the capacity of these yarns to support the stretching. It is significant to recall that during the weaving, the yarn is subjected to considerable stretching forces. The traction model developed in our previous studies is used to validate the feasibility of these new fibres integration in a woven structure. Also, it is necessary to compare them with other yarns already confirmed for our future application.
Traction model

The yarn traction behaviour is given by the elongation tests. This behaviour is obtained starting from the curve representing the strain exerted according to the stretching (Figure 3). In the textile area, and more particularly in the study of the yarn traction, several approaches have been developed (Vangheluwe 1993; Zurek and Aksan 1975; Manich 1999; Legrand 2000). The strongly nonlinear behaviour of the yarn is developed in particular. Starting from the results of these different works we set up the traction model trough parameters sensitivities of the various existing models (Bruniaux and Castelain 2003). Our model traduces the various stages of deformation of the yarn during traction. Each stage corresponds to a specific sub model.

\[ Y = \zeta \left[ 1 - (a + b \exp \left( -\frac{E}{c} \right)) \right] \left( E - E^* \right) + T^* \]

damaged. Beyond the point of passage \( E^{**} \), a progressive destruction of fibres starts, until the complete rupture of the yarn.

In this last case, the expression of the model is given below:

\[ Y = \zeta \left( E - E^* \right) + T^* \]

Identification of the model and result

The values of the parameters used for the model simulation are resulting from the identification with our samples. The superposition of the theoretical and practical curves shows that the selected model is well adapted to the study. Analysis of the mechanical properties shows that the PANI involvement on a PET spun fibre contributes to the improvement of traction rigidity opposing to those with a PET filament fibres. The value of \( \zeta \) in Table 3 confirms this result. Moreover, the elastic zone, given by the gap between \( E^{**} \) and \( E^* \) or \( T^{**} \) and \( T^* \), is larger due to the presence of PANI for PET spun fibres. In Figure 13 our samples are compared with other yarns usually used in the weaving industry. The stress/strain curves are located perfectly in the area where weaving is possible.

![Figure 3: Modelling of Stress-strain curves](image)

The first zone

The first zone corresponds to the setting in progressive tension of fibres. During this stage, fibres are subjected to longitudinal strengths which result in their alignment (Sabhi 1993). This zone corresponds also to fibres spatial organization towards deformation direction. The first zone is delimited by the parameter \( E^* \). Therefore, the first model is defined by the following expression:

\[ Y = \zeta \left[ \left( 1 - \frac{T^*}{E^*} \right) E + \frac{2T^*}{\zeta} - E^* \right] \left( \frac{E}{E^*} \right) + T_0 \]

The second zone

It represents the elastic zone of material. It is a linear one with the constant relationship between the strain and the stretching. The value of \( \zeta \) term is constant until a maximum limit beyond which the deformation is not elastic any more (plastic zone). The passage between the elastic zone and the plastic zone is identified by the point \( E^{**} \). The second model can be represented by the following expression:

The third zone

The last zone of the traction curve puts apparently nonlinear phenomena. In this zone, the fibres are strongly

![Figure 4: Stress-strain curves of original PET spun (▲), PANI/PET spun 6% (+), PANI/PET spun 10% (●) yarns and the model (solid line).](image)

Table 2: Identification results of model for different fibres

<table>
<thead>
<tr>
<th>No</th>
<th>( \zeta )</th>
<th>( T^* )</th>
<th>( E^* )</th>
<th>( E^{**} )</th>
<th>( T^{**} )</th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.638</td>
<td>1.096</td>
<td>0.968</td>
<td>2.964</td>
<td>2.399</td>
<td>1.833</td>
<td>-1.775</td>
<td>4.007</td>
</tr>
<tr>
<td>2</td>
<td>0.518</td>
<td>2.570</td>
<td>3.108</td>
<td>2.764</td>
<td>2.518</td>
<td>0.970</td>
<td>-1.160</td>
<td>3.656</td>
</tr>
<tr>
<td>3</td>
<td>1.024</td>
<td>3.955</td>
<td>2.824</td>
<td>2.103</td>
<td>3.700</td>
<td>1.237</td>
<td>-1.558</td>
<td>4.536</td>
</tr>
<tr>
<td>4</td>
<td>0.577</td>
<td>3.871</td>
<td>2.520</td>
<td>3.762</td>
<td>4.606</td>
<td>1.327</td>
<td>-1.679</td>
<td>2.605</td>
</tr>
<tr>
<td>5</td>
<td>0.093</td>
<td>2.345</td>
<td>2.922</td>
<td>3.700</td>
<td>1.600</td>
<td>0.138</td>
<td>1.220</td>
<td>3.000</td>
</tr>
<tr>
<td>6</td>
<td>0.363</td>
<td>3.557</td>
<td>2.730</td>
<td>3.678</td>
<td>3.896</td>
<td>0.953</td>
<td>0.050</td>
<td>2.297</td>
</tr>
</tbody>
</table>

where, 1 : Neat PET spun yarn (20 Nm), 2 : PANI/spun yarn 6%, 3 : PANI/spun yarn 10%, 4 : Neat PET filament
yarn t (292 dtex), 5: PANI/filament yarn 6%, 6: PANI/filament yarn 10%.

Figure 5: Stress-strain curves of neat PET filament (▲), PANI/PET filament yarn 6% (+), PANI/PET filament yarn 10% (●) and the model (solid line).

Table 3 shows the mechanical properties of PANI/PET conductive fibres compared to those of the neat PET fibres. The modulus of PANI/PET filament yarns decreases after a PANI coating from 6% of PANI/xylene solution. Although, the modulus of PANI/PET spun fibre yarn increases.

Table 3: Mechanical properties of neat PET and PANI/PET conductive fibres

<table>
<thead>
<tr>
<th>Yarns</th>
<th>Force of rupture (N)</th>
<th>Deformation of rupture (%)</th>
<th>Modulus (MPa)</th>
<th>Elongation (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neat PET spun fibre yarn</td>
<td>4.975</td>
<td>5.6</td>
<td>398462</td>
<td>5.61</td>
</tr>
<tr>
<td>PANI/spun yarn 10%</td>
<td>5.651</td>
<td>7.4</td>
<td>427556</td>
<td>7.34</td>
</tr>
<tr>
<td>Neat PET filament yarn</td>
<td>5.222</td>
<td>27.0</td>
<td>745102</td>
<td>18.01</td>
</tr>
<tr>
<td>PANI/filament yarn 10%</td>
<td>2.495</td>
<td>6.0</td>
<td>643552</td>
<td>4.44</td>
</tr>
</tbody>
</table>

These results can be explained twofold. First, the polar solvent (xylene) attacks to the polymer chains in amorphous region, which is very important to mechanical properties. Therefore, the mechanical properties of PANI/PET filament yarn decrease with the time in the presence of solvent. In the case of spun fibre yarns, it can be considered that the effect of the formation force between the inter-fibres is more important to that of solvent at amorphous region. As our PANI/PET fibres show the good mechanical properties, these fibres can be transformed in textile structures by weaving, knitting or other manufacturing process.

CONCLUSION

In this study, PANI coated PET conductive fibres prepared by coating processes. Electrical and morphological properties have been studied in the first section of paper. PANI coated PET conductive yarns show good electrical properties preserving their original strength and flexibility. In the second part of paper electrical and mechanical model of conductive yarns are presented. These models appear independently. However they are closely linked and these researches are currently carried out in our laboratory in order to obtain relation between the electrical and mechanical properties of conductive polymer fibre yarns. The important conclusion of mechanical modelling is that conductive yarns remain usable in textile processes.

References


Legrand X., 2000 Contribution à la modélisation dynamique et à l’identification de tissus techniques, Theis Université des Sciences et Technologies de LILLE.


Vangheluwe L., 1993 “Relaxation and inververse relaxation of yarns after dynamic loading” Textile Research Journal; 63(9), 552-556.


FIBRE TEXTILE APPLICATIONS
GEOMETRIC PRE-PROCESSOR FOR THE CALCULATION OF PHYSICAL PROPERTIES OF TEXTILES

Martin Sherburn, Francois Robitaille, Andrew Long and Christopher Rudd
School of Mechanical, Materials and Manufacturing Engineering
University of Nottingham, University Park
Nottingham NG7 2RD, United Kingdom
E-mail: emxmns@nottingham.ac.uk

ABSTRACT

Predictive simulations of the manufacturing processes for composites and calculations of the performance of the resulting parts require geometric models of the textiles used. These models can be created manually with the help of commercial 3D modelling software. However this is very time-consuming, error-prone and inefficient. Hence the need for 3D modelling software that applies specifically to textiles, and is general enough to be able to model any type of textile. Such software is currently under development by the present authors and is the focus of this paper. Although originally designed for the modelling of composites, this tool has many other applications.

INTRODUCTION

Modelling of textiles is important for the production of composites. In order to produce composites at low cost, it is necessary to have a quick development process, good reliability and good predictability. Hence fast and accurate modelling of the manufacturing process is required. Ideally one would want to model the geometry of the whole textile preform, which is the textile backbone of a composite part, at the detail of the individual tows. However this is not realistic; modern computers are not sufficiently powerful to perform such calculations within a reasonable time-frame. Instead, the most common approach is to model the preform as a shell surface with different local properties. For example, the injection of resin through a textile preform in a resin transfer moulding mould can be modelled with software such as LIMS™ (Simacek et al. 1998) or PAM-RTM™ (ESI Group 2002) (Figure 1). The local properties of the preform such as its permeability can be measured experimentally, but comprehensive data measurements are hard to obtain. Alternatively, the local properties can be obtained through simulations, hence the need for geometric models of the textiles.

Since the analysis of all the tows in the textile preform is too computationally expensive to perform, only a representative small section of the preform defined at the level of the tows is analysed. Furthermore the analysis of each individual fibre in the tows contained within that small section is computationally expensive although feasible (Zhou et al. 2004). Hence in the interest of speed, the tows are modelled as solid volumes with transverse isotropic properties. The properties of the tows can be determined either using analytical models or may be calculated from computer simulations.

Figure 1: Injection of resin through a textile preform in an RTM mould

This paper introduces a general formalism for the efficient geometrical description of all textiles. The formalism is fully implemented in a software package TexGen developed by the authors.

Modelling the mechanical properties of the final composite part is also important in order to be able to predict its performance. The procedure of simplifying the geometry down to levels that can be analysed is applied in the same manner to the modelling of the mechanical properties. The difference lies in the calculation techniques and software employed. In this case commercial finite element analysis software such as Abaqus™ (Abaqus Inc. 2003) can be used but is not the only option.

The geometric model presented here need not be restricted only to analysis of the manufacturing and performance of composites. There are many potential applications including heat transfer across textiles for winter garments, simulations for chemical and fire protective clothing and mechanical properties of textiles for geotextile applications.

DESCRIPTION

TexGen is able to model all textiles using the same unified format. The format consists in describing the tow paths with the use of vectors (Robitaille 1999-2002) for a repeatable unit cell. Only a small number of vectors are needed, typically one vector per tow cross-over. An in-plane shear deformation may be applied to the vectors at this stage if required; furthermore vector randomisation may be applied to simulate variability in the textile. The
vector paths are then smoothed using piecewise cubic Bézier curves. In order to give the tows a volume one is able to assign an arbitrary cross-sectional shape to the vectors. Any interferences between the tow volumes must then be corrected before exporting the model. Each of these steps is briefly described below.

**TOW PATHS**

The geometry of many un-deformed textiles can be defined as a unit cell of tows which repeats itself in two orthogonal directions. It is therefore possible to define an infinitely large textile simply by specifying the geometry within the unit cell and the repeat distances. The geometry of the tows within the unit cell is defined by the paths of their centrelines. Each path is defined by a list of points given in 3D space; these points are then connected together by vectors (Robitaille 1999-2002). A simple case of a plain woven textile is shown in Figure 2 viewed from above hence the z coordinate is not visible. In order to conserve continuity each end of each tow in the unit cell must join together with its other end in the repeated unit cell. The unit cell consists of 4 tows described by 8 vectors. The repeat distance in the x direction is a, the repeat distance in the y direction is b.

**IN-PLANE SHEAR DEFORMATION**

When draping a textile over a mould, in-plane shear will inevitably occur. It is therefore necessary to simulate the effect of shear. For textiles where all of the tows are parallel or perpendicular to each other this can be achieved by transforming the vectors that describe the path of the textile in such a way that the lengths of the vectors do not change. To ensure that the lengths do not change one can imagine that the textile is clamped within a square frame as shown in Figure 4, making sure the tows are parallel to the frame.

Figure 2: Example of Vector Paths for a Plain Weave

Figure 3 shows the path of vectors for a weft knit textile. The repeat distance ‘a’ ensures that the tow joins up with itself after being repeated. The repeat distance ‘b’ is shorter than the height of the unit cell; this is because the tow in the unit cell crosses over with tows in the repeated unit cells. The unit cell consists of just 1 tow made up of 13 vectors.

Figure 3: Example of Vector Paths for a Weft Knit

Figure 4: Shearing of a Square Frame
The frame is sheared while keeping the length of the edges constant. To determine the new position of any point within the original square a simple linear transform can be used. The x coordinate of the original point is multiplied by \( x' \) while the y coordinate is multiplied by \( y' \). The values of \( x' \) and \( y' \) depend on the amount of shear imposed on the frame (i.e. the angle \( \theta \) after shear) as shown in Equations 1a and 1b.

\[
x' = \sqrt{2} \cos \left( \frac{\theta}{2} \right) \tag{1a}
\]
\[
y' = \sqrt{2} \sin \left( \frac{\theta}{2} \right) \tag{1b}
\]

As mentioned above the taws must be parallel to the frame edges. If they are not then the lengths of the taws will not remain constant after the transformation. Realistic shearing of taws that are not perpendicular to each other has not yet been implemented in TexGen. The result of shearing a 2x2 twill weave can be seen in Figure 5.

![Figure 5: Shear of 2x2 Twill Weave](image)

**VECTOR RANDOMISATION**

The inherent variability within textiles has a big impact on their mechanical properties; it is therefore important to model this effect. This can be achieved by adding variability to the paths of the taws either by manually incorporating it into the vector path input files or by taking advantage of the in-built vector randomisation routines.

The automatic approach consists in moving the ends of the vectors by a random distance in a random direction. Connecting vector ends are moved by the same amount in order to avoid discontinuity along the tow paths. In addition, vector ends that are directly above or below each other are kept that way to ensure that the peak of the tow path curve occurs at the tow cross-over.

A simple randomisation distance can be specified between zero and some arbitrary maximum. Or a more sophisticated method can be used involving a cumulative inverse normal distribution function. Given a standard deviation and a completely random value between zero and one (where the probabilities of obtaining any of the values within the given range are equal); a random number that follows a normal distribution can be obtained from the cumulative inverse normal distribution function (Figure 6).

![Figure 6: The Inverse Normal Cumulative Distribution Function](image)

**INTERFERENCE**

It is not always straightforward to specify tow paths in such a way that no interference between the envelopes will be present. Interference occurs when the volumes of two taws intersect or when a tow volume intersects with itself. If simulations are performed while interferences exist the results will be inaccurate or the simulations will fail completely. It is therefore vital to be able to detect such interferences and correct them.

The tow surfaces are constructed from a number of vertices connected together with triangles. The triangles form a completely closed volume, so in order to penetrate a tow volume it would be necessary to penetrate at least one of its triangles. Hence the interference detection method consists in searching for intersections between triangles (Bourke 1997) (the number of checks performed is approximately equal to the number of triangles squared). The intersections between adjacent triangles are ignored since they will always intersect at their common edges/corners. If any other intersections are found then it is safe to assume that
the volumes of the tows to which those triangles belong intersect.

There is one flaw with this method however: if the volume of one tow is completely contained within the volume of another then the surfaces of these two tows will not intersect. Since the interference detection technique relies on the surfaces intersecting, this method will fail. Such cases would not happen in practice and hence were deemed un-important.

Detecting interferences is only the first step since it is desirable to be able correct them automatically. As mentioned before, the surfaces of the tows are constructed of vertices connected by triangles. From the interference detection method, the vertices of the triangles which are interfering can be identified. Each vertex belongs to a node along the centreline of the tow, so a vector pointing from the vertex to the node it belongs to can be drawn. This vector represents the direction in which the node must be moved to reduce the interference. If several vertices causing interference belong to the same node, then the average of these vectors is taken.

Figure 8 illustrates the procedure of correcting interference in 2D. The two polygons represent cross-sections of two intersecting tows. Here the surface is represented by lines rather than triangles since it is 2D. Two lines from each polygon are intersecting with each other; vectors are drawn from the vertices of these lines towards the centre node. The average of these vectors is shown by the dotted vector; this is the direction by which the tows will move. There is no simple analytical method to find out how far the node must be moved in order to correct the interference therefore an iterative approach is used. The node is moved by an arbitrarily small amount, the model is re-created and then interferences for the new model are detected. This process is repeated until no interferences are found.

Moving individual nodes on their own tends to create a jagged shaped tow. In order to avoid this, adjacent nodes are also moved in the same direction but by a smaller distance. The fraction of the distance by which it is moved depends on the distance between the nodes and is calculated as shown in Equation 2.

\[ F = \frac{1 + \cos \left( \frac{\pi}{d_{\text{max}}} d \right)}{2} \] (2)

where \( d \) is the distance between the nodes, and \( d_{\text{max}} \) is the maximum distance by which the nodes will still be affected. Any nodes past the distance \( d_{\text{max}} \) do not get moved. Figure 9 shows the result of the interference algorithm for a simple case with two perpendicular tows.

![Figure 9: (left) Interference Detected Shown in White (right) Interference Corrected](image)

The interference correction method presented here is best suited for cases with limited interference.

**MESH GENERATION AND OTHER OUTPUT**

TexGen does not create meshes from the geometric model; instead it relies on powerful third party software such as Gambit™ (Fluent Inc. 2003). Gambit™ journal files can be created which provide all the data necessary to mesh the model. There are two basic types of elements that can be used, either tetrahedral elements or hexahedral elements, each of which can contain different number of nodes (4/10 and 8/20/27 respectively). Tetrahedral elements are suitable for the tow volume as well as the resin volumes, whereas the hexahedral elements are only suitable for the tow volumes. If a mesh is required for both the resin volume and the tow volumes then it is necessary to use tetrahedral
elements for the entire mesh. This is because hexahedral elements do not tessellate with tetrahedral elements.

Output for other calculation methods is also available: files can be created to be used with the stream surface approach (Robitaille 2003) or 2d grid method in order to calculate permeability.

TOW ORIENTATIONS

Tows have different mechanical properties in different directions so it is important for finite element packages to know the orientations of the tows. Since the mesh is not created within TexGen, it has to be imported from third party mesh generators. The orientation of each element is determined by finding the closest tow vector to the centre of each element. The element direction is then set equal to the direction of this vector. With these orientations calculated, TexGen is able to create input files that are compatible with Abaqus™ (Abaqus Inc. 2003) and potentially any other finite element software package.

APPLICATION EXAMPLES

Figure 10 shows a weft knit model generated and viewed within TexGen. Thirteen vectors were used to describe part of the path of one tow; the other tows were generated automatically by copying this small section displaced by the repeat distances. Excess tow geometry found outside the unit cell is trimmed and surfaces are added to fill the gaps where the tows where cut.

Figure 11 shows the mesh for a braided textile where the geometry was generated by TexGen and exported to Gambit™ (Fluent Inc. 2003). All the information necessary to re-create the geometry in Gambit and perform the meshing is read from the exported journal file.

Once the mesh is created, the orientations of the elements may be calculated. Figure 12 shows a graphical representation of the orientations, for each element a line segment is drawn at the centre of the element pointing in the direction of the orientation assigned to it.

Figure 13 shows a stress/strain analysis using the finite element package Abaqus™ (Abaqus Inc. 2003) on a 2x2 twill weave composite under tension in the 1 direction. Two separate images are provided, one showing the stresses in the resin volume while the other shows the stresses in the tows. Note that the scales for the contours are not the same in both cases, the stresses in the tows are much higher than those in the resin.

TexGen is also able to create slices across a model to get a 2D section containing the tow shapes. This can be used with commercial finite element packages such as Fluent™ (Fluent Inc. 2003) or using alternative methods such as the stream surface approach (Robitaille 2002) to predict permeability (Figure 14).

The algorithm for automatically generating 2.5D stream surfaces (Robitaille 2003) is also implemented (Figure 15). The output from this can then be used to calculate permeability values for the unit cell.
Figure 11: Braid Mesh Generated by Gambit™

Figure 12: Element Orientations for Weft Knit Textile
Figure 13: Stress Distribution Contour Plot for a 2x2 Twill Weave Composite Generated by Abaqus™

Figure 14: Slice of a Plain Weave Textile
CONCLUSION

In this paper the implementation of a method of representing the geometry of any textile which can be used for calculations is introduced. Shear deformation of the resulting models can be simulated for cases where the tows are all parallel or perpendicular to each other. Randomness of the tow paths can be introduced to simulate real textiles with the use of a normal distribution. Interference between tows can be corrected automatically avoiding difficulties in subsequent calculations. The final model can then be used for a large number of different applications such as calculation of mechanical, flow and heat transfer properties.

REFERENCES

ABAQUS Inc. 2003 Abaqus/Standard user’s manual v6.3.
SIMSERV
HOW TO STIMULATE WIDER USE OF SIMULATION IN EUROPEAN INDUSTRY? THE ANSWER OF SIM-SERV

Johannes Krauth  
SDZ GmbH  
Hauert 20  
44227 Dortmund, Germany  
Johannes.Krauth@sim-serv.com

Kaj Justlin  
VTT Industrial Systems  
Tekniikantie 12,  
Espoo, Finland  
Kaj.Justlin@vtt.fi

ABSTRACT

This paper introduces the virtual institute on production-oriented simulation Sim-Serv (www.sim-serv-com) which started operating with financial support by the European Union two years ago. The paper describes Sim-Serv’s objectives, its organisation and the services it offers to researchers as well as to commercial suppliers and users of simulation tools and services.

INTRODUCTION:VIRTUAL INSTITUTES

Sim-Serv has been set up with financial support by the European Union in its Fifth Framework Programme for Research and Development. It is one of currently 17 Virtual Institutes which focus on different areas of technology. The mission of these virtual institutes is to stimulate and coordinate application oriented research and development in dedicated areas, and to ensure a smooth transition of results into applications in industry and society. Each Virtual Institute (VI) is financed by the EC for a limited period, but is meant to operate beyond this period, hence each VI will establish a self-financing organisation which will carry on the activities at the end of the EU funded project.

The VI Sim-Serv (www.sim-serv.com) focuses on product- and production-oriented simulation. It provides practical support to researchers and developers, to those offering tools and services on a commercial basis as well as to industrial users.

THE OBJECTIVES OF SIM-SERV

1. Stimulate the Application of Simulation in European Industry

It is commonly accepted that simulation – in spite of its obvious power and benefits – is not widely used in industry, clearly not as wide as it should be. The estimated potential for savings and improvement in European industry, which could be achieved by proper use of simulation is enormous. And those companies who used it express a high degree of satisfaction. In the nineties, simulation was considered a rapidly growing market, and even the American Integrated Manufacturing Technology Initiative (www.imti21.org) initiative ranked simulation as one of the four most important technologies for future manufacturing. It states that "no other approach offers more potential for improving products, perfecting processes, reducing design-to-manufacture times, and reducing product realisation costs".

In reality however, the growth of simulation was rather slow. A study of the use of system simulation in UK manufacturing industry (Hollocks, 1992) showed a 92% satisfaction rate amongst users, yet a penetration into industry of less than 10%. That pattern is reported as common across Europe (Kerckhoff, 1997). A more recent study shows that not much has changed in the last decade (Abdel-Malek, 1999).

Why is this so?

Our hypothesis is: Simulation and its benefits are still insufficiently known in industry, especially among those who make the decisions whether to use it or not (Wisniewski, 1994). A second problem is: simulation is difficult to justify from an economic point of view. Simulation is often used to prove the viability of a system design. But what is the added value of such a proof in terms of money? Maybe the model helps improve the design, but maybe the improvement is also possible without simulation? It is very difficult to make a convincing financial case for simulation.

The main objective of Sim-Serv is to stimulate a wider use of simulation technology in European industry and thus help European companies (especially small and medium-sized enterprises) meet the challenges of global competition.
If our hypothesis is correct, then this requires spreading information about simulation to industry managers, and providing detailed information on the benefits achieved by other users. It requires gathering such meta-knowledge, and applying it to specific customer requirements.

Therefore, Sim-Serv’s main activity is dissemination: providing a central entry point for those seeking information, guidance and support. Newcomers find general information, case studies about successful applications (with an emphasis on the business dimension), links to experts, suppliers and tools as well as a help desk ready to answer any questions regarding the use of simulation.

In addition, industrial users find independent consultation and advice about the possibilities of applying simulation technology to their business, the expected cost and benefits. An in-depth analysis of the problem carried out by neutral specialists leads to a recommendation of suitable solutions and of suppliers able to deliver them.

2. Support European Simulation Suppliers

In Europe, the supply side consists of numerous small or even micro-enterprises offering in many cases highly specialised tools and solutions. Many of them are recently born spring-offs from research institutes. They tend to have problems getting Europe-wide visibility and finding customers from a broad range of industry sectors. Through its wide dissemination and marketing activities, Sim-Serv facilitates access to a European market even for small or micro providers.

Of course the above mentioned dissemination activities also support European simulation suppliers in that they help expand the market for their tools and services. Sim-Serv is particularly keen on spreading information about innovative techniques and tools, new application areas etc.

3. Facilitate Joint Research and Development

Sim-Serv also stimulates co-operation and joint developments of suppliers. Sim-Serv supports its members by bringing together partners with similar or complementary aims and skills, and helping them form consortia. Sim-Serv supports the development of research projects and the application for research funding. On the long run Sim-Serv aims at the development of widely accepted standards and quality criteria as well as a suite of tools, which comply the standards and criteria and are offered under a common brand.

Finally, Sim-Serv provides an overview of the state of the art in simulation, it identifies trends and unsolved problems, and thus provides guidance for research and - last not least - research policy. Sim-Serv will play an active role also in future EU research programmes.

ORGANISATION

The basic idea of Sim-Serv is to act as a neutral “mediator” facilitating a smooth matching and interaction of demand and supply. Wherever possible and reasonable, this interaction should use electronic media, however it is understood that face to face meetings cannot always be replaced by virtual interaction. Hence local presence is as essential as European wide recognition.

Sim-Serv is currently lead by a Core Team consisting of three partners:

- VTT Technical Research Centre of Finland, Systems Dynamics Group (Espoo, FI, www.vtt.fi)
- Simulations-Dienstleistungs-Zentrum SDZ (Dortmund, D, www.sdz.de)
- City University, Centre for Systems and Modelling, Manufacturing Support Unit (London, UK)

This Core Team is responsible for the European wide recognition and for development of the network. It is locally supported by a network of Local Contact Points. Local Contact Points provide information and services in local languages and are available for face to face meetings whenever the need comes up.

The Sim-Serv Supplier Group consists of currently 40 members, the number is steadily growing. They represent a good mix of complementary skills and cover the majority of EU member states plus some Central European countries. At present there is almost a balance of academic and commercial partners in the group.

The role of suppliers is to deliver customised solutions on request on commercial terms, and to provide input to the institute’s web site. Whenever the Core Team has received a request from an industrial user, it forwards this request to the Supplier Group. By an internal bidding mechanism (see figure 2 below) the best supplier is then identified and brought in touch with the user.

All suppliers are presented on the institute’s web site.

The Supplier Group is open to new members at any time. It is indeed one of the aims of Sim-Serv to expand this group substantially.

In addition, Working Groups (WGs) are being set up dealing with various technical or commercial issues of
common interest. WG members may be suppliers or users of simulation tools and services, members or non-members of the Sim-Serv network. Also WGs generated elsewhere are invited to use Sim-Serv’s facilities and make themselves known via Sim-Serv.

Sim-Serv’s Core Team and the activities of the Local Contact Points are currently funded by the European Union. In order to continue the activities after the end of the funded project in October 2004, the Core Team has founded the not for profit Sim-Serv Association based in Finland. This association is open for all organisations involved in developing, selling or using simulation technology in Europe.

**SIM-SERV’S SERVICES TO INDUSTRY**

The central service of Sim-Serv is the web site. It contains general information about simulation, a database of technical and scientific information, news and information on relevant events, case studies and success stories, a list of suppliers and links to them as well as to other sites of interest, e.g., to European simulation societies. The working groups find their place on the web site, and reports about test cases can also be found.

A relatively new service is the self evaluation tool, which is freely available on our web site. A simple questionnaire of 10-12 questions helps an industrial user define very roughly his situation and aims, and to check if simulation should be applied in this particular situation. 2-3 days after submitting the questionnaire, he will receive a report and a recommendation written by one of our experts.

Besides, Sim-Serv offers the following to potential simulation users in industry (figure 1):

- the web site www.sim-serv.com contains general information on simulation, success stories, information about suppliers and relevant links
- working groups provide a forum to express user problems and needs
- the help desk answers specific questions and offers a first and rough evaluation of problems
- independent technical consultation supports the user in analysing his problem and checks if simulation should be applied and how,
- a supplier-independent functional specification of the application/solution is developed on request by technical experts
- the best suited supplier(s) of the specified solution are found (see below for details)

- project management support, quality assurance and an evaluation of the solution after its implementation are offered

![Sim-Serv Procedures](image)

**Figure 1: Services to Industrial Users**

**CASE STUDY EXAMPLES**

In a number of test cases, the Sim-Serv approach proved feasible and beneficial to both customers and suppliers. Test cases are industrial applications of simulation where Sim-Serv assisted the customer analyse the problem, checked applicability of simulation and searched for suitable suppliers.

A medium sized UK based manufacturer of aluminium parts was looking for tools to support scheduling. Sim-Serv established a contact to Riga Technical University who developed a simulation model of the plant and used it for testing out several scheduling strategies and predicting the effects of introducing a particular tool. As a result, they managed to reduce stocks of raw material by 50% (Merkurieva, 2003, 2004).

A Finnish manufacturer of rubber and plastic parts had to reduce product development time. With the support of Sim-Serv they tested a simulation tool and decided to introduce it. Production cost was reduced by 30% on average (Ture, 2003).

A French manufacturer of laminates was offered a new technology for producing multi-ply laminates. They were looking for a proof of technical and economic feasibility of the proposed process. Sim-Serv brought them in touch with two simulation service suppliers: one of them developed a model of the entire process which allowed to estimate production output and cost. The second made a very detailed model of the most critical part of the process in order to prove technical feasibility (Mallinson, 2004).
A study on an application in a German manufacturer who had to design a new assembly line is described in this volume (Krauth, 2004).

SERVICES TO RESEARCHERS AND SIMULATION PROFESSIONALS

To researchers and simulation professional (suppliers), Sim-Serv offers essentially two advantages:

- the chance to present themselves to potential users in industry and thus to find additional partners / customers
- the chance to network and co-operate with other simulationists, to co-ordinate and join forces.

More specifically, the following is offered:

- space on our web site to present themselves, their expertise, their successful projects, and their tools
- guidance and support for the preparation of this material
- support for its translation to other European languages
- news and information about relevant events on the web site
- a data base containing up to date technical information
- working groups as a possibility to co-operate with other suppliers
- the chance to contribute to joint (funded) research and development projects
- professional, European- wide dissemination activities to attract potential users to our web site, and to acquire commercial or research projects
- offerings for additional commercial projects acquired by the Core Team

Sim-Serv helps its members find partners/customers, and it searches customers itself, e.g. by means of the self evaluation tool. Whenever a customer approaches Sim-Serv and asks for some simulation service, Sim-Serv offers the services described above: neutral advice, supplier independent development of a functional specification, search for suited supplier(s). The search for suppliers is done by the following “internal bidding procedure”:

This specification and an Invitation to Tender are then circulated to the Suppliers Group or to a subset of members pre-selected jointly by Sim-Serv and the customer.

The members who are interested in the offered projects submit their offers, which contains details of the solution they offer, the price and the earliest possible delivery date.

Based on criteria defined by the customer, Sim-Serv evaluates these offers and presents them to the customer who makes a final selection.

![Sim-Serv Procedures (2)](image)

In reality, some iterations may be needed, e.g. suppliers may ask for more information before they submit an offer, or the customer asks for modifications of the offers. In general our experience shows that this procedure is considered effective and fair by both sides.

SIM-SERV’S FUTURE

Sim-Serv as a funded project will terminate end of October 2004. From then on the Sim-Serv Association will continue providing the services and operating the web site.

Sim-Serv closely co-operates with other virtual institutes, mainly with the virtual institute for advanced manufacturing technologies ADMAN (www.max-serv.com). These two institutes seem to complement each other in a most natural way.

The Sim-Serv Core Team is aware of the existence of numerous organisations active in the simulation field. The intention is by no means to compete with them. We rather intend to complement the more science-oriented organisations such as the national simulation societies or EuroSim, and to support the commercial organisations in order to promote our common goal.
To improve the general knowledge about simulation and its benefits, particularly in industry, to stimulate and facilitate a wider take-up, and to create an environment for fruitful and exciting further developments of simulation technologies.

All researchers, commercial suppliers and users of simulation technology are invited to join Sim-Serv, use our services and become members of the Association in order to contribute to the shaping of the future of simulation in Europe.

For more information, please contact:
Dr. Johannes Krauth
Sim-Serv Services and Quality Manager
Adolf-Reichwein-Str. 32, D-28329 Bremen
phone +49.421-437 3676
e-mail: Johannes.Krauth@sim-serv.com
www.sim-serv.com

REFERENCES


Wisniewski M, Jones C, Kristensen K, Madsen H and Ostergaard P (1994) “Does Anyone Use the Techniques We Teach?”, OR Insight, 7, 2, 2-7, April-June

JOHANNES KRAUTH was born in 1951. He studied mathematics, computer sciences and political sciences. He received his degree in mathematics on 1997 in Berlin, and his doctorate in Vienna in 1990. He was working in several research institutes and is involved with simulation since 25 years. He is one of the initiators of Sim-Serv.
Experiences from Sim-Serv’s Project Work: 
The Kaeser Case

Wilfried Krug, Sven Pullwitt1 
Dualis GmbH Dresden 
Johannes Krauth 
Sim-Serv / SDZ GmbH Dortmund2

Abstract

The paper describes an industrial application of discrete event simulation: The design of a new production line is checked and optimised. From a technical point of view, this is a standard application, therefore little is said on the technical side of the project. The interaction between the customer, the Sim-Serv team and the chosen supplier is described in detail. It is shown that the approach of Sim-Serv can help even small and medium-sized companies getting started with simulation technology: The medium sized company Kaeser has turned from scepticism and reservation to daily use of simulation.

1. Introduction, Background

Kaeser Kompressoren is a well established medium-sized enterprise in Coburg, Germany. They have decades of tradition in development and manufacture of air compressors for such diverse applications as refrigeration, air conditioning, or petrol production.

Changing market requirements force them to adapt and improve their products and production processes permanently. Kaeaser’s engineers therefore possess substantial experience in designing production systems, and they are using different concepts. Picture 1 shows two of their products. The one on the left is produced on a so-called “splitted line” where a variety of different products can be produced, the one on the right is produced on a “compact line” dedicated to one type of product only.

Due to the globalisation of markets, development and production have become more and more order driven. Therefore Kaeaser Management searches information technologies to support the transformation and management of their production.

To date simulation and mathematical optimisation had never been used inside Kaeaser. However some of their engineers had theoretical knowledge about these technologies, their power and the benefits of using them. Since several years, Sim-Serv and its ancestors had sent periodically new information about simulation technology and its potential.

In the year 2001, a new product was developed at Kaeaser. For its production, a new assembly line was to be designed. A major design goal of this new line was; Throughput times of the new product should be substantially shorter than before. For this project, the planning team seriously considered using simulation to evaluate and optimise the design.

However, there was no clarity as to how and when simulation could best support the planning process. Therefore they started as usual and developed a detailed layout. Only when some open questions which could not be answered easily, stopped the progress of the planning process, they decided to simulation specialists.

1 Dualis GmbH IT Solutions, Tiergartenstr. 32, D-01219 Dresden, www.dualis-it.de
2 Sim-Serv: European Virtual Institute for Production Oriented Simulation Services, www.sim-serv.com
This paper is dated April 22nd, 2004

494
This decision was facilitated by Sim-Serv, an EU-funded institute which offered neutral and competent support.

Sim-Serv is one of the “Virtual Institutes” which have recently been initiated by the EU. The objectives of these virtual institutes are to disseminate information about the state of research and application in a particular field of technology, and thus to make this technology easily accessible to newcomers. To this end, the virtual institutes run web sites, publish in newspapers and magazines, contribute to conferences and organise seminars and workshops on their respective technology. In addition, they offer individual consulting and support for first time users.

2. Project Preparations: The Role of Sim-Serv

The technology represented by Sim-Serv is simulation, especially its application in industrial production. In the case describe here, Sim-Serv provided the following services:

First, the company Kaeser was assisted during the analysis of the problem. Several discussions between Kaeser’s team and Sim-Serv helped clarify how simulation could contribute to solving the problem at hand, and if its use would pay back.

In principle, a detailed calculation of cost and benefits is not possible in the case of planning a new system because the expenses of the system, and particularly the savings achieved by using simulation, are not known beforehand. Only ex post can the expected and the real expenses be compared. The advantage of using simulation as a support for designing new systems is mainly that it improves substantially the information available for making decision. The effects of alternative solutions on target parameters and costs can be calculated quite accurately.
This advantage finally convinced the planning team of Kaeser, and it was decided to make a first try with simulation technology.

As a second step, Sim-Serv and Kaeser’s planning team developed jointly a specification of the simulation project. Due to the little knowledge they had about simulation, the engineers of Kaeser were unable to produce this specification without Sim-Serv’s support.

After some additional non-technical constraints had been clarified (e.g. regarding the due dates and the communication language), Sim-Serv went off to look for suitable suppliers of the requested simulation service. To this end, the specification was sent to approx. 20 experienced service suppliers, and they were invited to tender. Within a week several offers were received. Based on the criteria defined by Kaeser, Sim-Serv ranked these offers and gave a recommendation. The offer of Dualis GmbH seemed to fit best to the user’s expectations.

The planning team agreed to this recommendation. As a next step, they now had to get the agreement of the CEO. He had to sign a contract with Dualis. After several months of internal discussions, he finally agreed to do so. The following factors were of particular relevance for this decision:

- the impressive and convincing presentations given by Dualis at Kaeser’s – they demonstrated clearly the benefits of simulation from a CEO’s point of view
- the range of experience from earlier projects Dualis could refer to, as well as the many success stories available on the Sim-Serv web site which proved that simulation was a well established technology offering substantial economic benefits
- the quality assurance which was given by the technical and organisational preparation by Sim-Serv and by their supervision of the entire project
- last and probably least the reduced financial risk: part of the project expenses were covered by some EU funding which Sim-Serv could offer.

Now that the engineers, planners and also the management of Kaeser had sufficient information about the possibilities and benefits of simulation, a contract was signed with Dualis with the following content:

- in a first project step, Dualis will build a simulation model of the new line. This model will enable the planning team to check if the designed system can produce the requested output and where improvements or savings can be made to the design. As a constraint for design and modelling, the shop floor layout as presented in picture 2 was given. The new line was to be integrated as a “compact line” in the centre of the existing production building.
- In case this step would lead to satisfactory results and Kaeser were convinced of the benefits of simulation, the model of the new line should be used in the future for planning and controlling the production on the new line. Preparations to enable this future use had to be made already during the first step.

3. Project Execution

As a first step – before model development – possible variations of the existing design were defined. In a discussion with all people involved in the new process (engineers, production managers, foremen) the variations were assessed in the light of the objectives. They were then ranked, and the steps and due dates for modelling these variations were fixed.
Then Dualis developed a model of the production process using their proprietary simulation software SPEEDSIM (see picture 3). Dualis selected the SPEEDSIM software for this project because it supports very fast model building, it is very easy to use even for persons with no prior knowledge of simulation, and it is particularly well suited for use in the foreseen project stage 2: The software is extremely cheap and easy to integrate with an optimiser and other software applications.

In the SPEEDSIM model, the relatively complex layout of the new line, as shown in picture 2, was represented by very simple graphics, as shown in the left part of picture 3. Initially, Kaeser staff was sceptical and did not believe that such an abstract and simplified model could ever produce any reasonable and relevant results. However, when the simulation runs produced results which coincided with their experience from the past, their confidence grew slowly. After only two days, they knew that by using simulation much faster and much better results can be achieved than they ever could achieve using their earlier planning methods. For example, by experimenting with the model, they found a practical way to reduce throughput times and at the same time reduce buffer capacities without any extra equipment or workers.

---

3 The new line consists of the two white areas in the middle of the picture. Red elements belong to the main flow, yellow elements are sub-assembly lines or warehouses feeding into the main line.
Picture 3: The SPEEDSIM simulation model of the new line and some simulation results\(^4\)

Once the initial scepticism had been overcome and the planning team had understood how easily and fast a wide range of model variations can be simulated and analysed with respect to all kinds of performance parameters, they generated manifold new ideas for model improvements and in depth investigations which had never been possible without the support of simulation technology. The simplicity of the SPEEDSIM model helped make the engineers of Kaeser understand the tool and its use, and it enabled them to do simulation experiments on their own.

As an example of a new idea, the effect of special handling devices was studied. It was found that they helped reduce the space needed (a critical problem in this project) without a negative effect on any of the other performance parameters.

The virtually infinite number of possible model variations led to the idea of using an automatic optimisation tool for finding the optimal parameter configuration. This idea was not applied in the planning stage because such an automatic search cannot invent new ideas, but it will be used in the second project stage when production planning and scheduling will be supported by an optimisation tool. This will be done by coupling the two Dualis software tools SPEEDSIM and ISSOP /KRU1/.

---

\(^4\) The left hand side shows the model of the line, the diagrams on the right show the utilisation of several work places over time. It is easy to see that the stations “Prüfen” (testing) and “Verkleiden” (assembly of housing” are underloaded.

\(^5\) Rapid developments in technology made it easy to apply simulation and optimization in the day - to - day operation of industry and commerce. By improving the decision base, the combination of these two technologies helps management to avoid errors and to minimise entrepreneurial risks.

ISSOP is an Intelligent System for Simulation and Optimization with seven nonlinear mathematical strategies in parallel. An intelligent learning process is integrated into this tool.
4. Project Results

Project stage 1 was finished in time and with good success. Nowadays the new line is in operation and satisfies all requirements defined at project start. The involved engineers are convinced that such a robust and elegant solution had not been found without the experiments on the simulation model.

A huge number of model variants had been analysed to achieve this result: The degree of automation had been varied as well as the position of machines or the position and size of buffers. Different organisational options had been compared: a straight line versus a tree structure, where several side lines feed sub-assemblies into the main line. New ideas such as the one mentioned above were introduced. All these model variants were tested under different conditions, using different loads of orders. Total production output, throughput times and setup times were the main performance parameters to compare the many variants. Due to the many customer specific product variations, the result however is not a pure “compact line”. The initial intention of having a production line which could be operated without any setup operations, was soon found unrealistic.

Besides a considerable reduction of design time (8 months instead of the 2 years that were expected initially), Kaeser now enjoys throughput times 8% less than foreseen, reduction of buffer capacities by 12% as compared to the initial design, and 6% less investment cost. Taking into account the quantitative benefits only, one can conclude that the extra investment for using simulation will pay back in less than six months.

Kaeser is now preparing for the second step: the simulation model will be integrated with the optimisation tool ISSOP and with the existing ERP system. As of October 2004, this integrated tool set will be used for production scheduling on the new line. The architecture of this integrated toolset is outlined in picture 4.

Due to the support offered by Sim-Serv, Kaeser made a big step forward towards full integration of information and production technology. This has and will continue having a positive influence on their competitive position.

5. Summary

The effective support and preparation by Sim-Serv was an essential factor for the success of this project in which a medium sized enterprise overcame its scepticism with regard to simulation and learned how to use this technology for improving their performance. The excellent service delivered by Dualis, the supplier which was recommended by Sim-Serv, proved that simulation can be very useful also in medium sized companies. “Without the neutral support by Sim-Serv, maybe we’d never dared to use simulation for such an important project” says Kaeser’s production manager Bernhard Brand, “however it was worth trying. We identified a number of important improvements to the new line. We will continue using the model in the future to support production scheduling and control.” This project again confirmed the approach of Sim-Serv: intensive information and customer oriented support helps expand the community of simulation users even into the domain of SMEs. This is to the benefit of all involved parties: The user companies gain access to new tools and improve their competitive position, the suppliers of simulation tools and services find new customers and thus gain a stronger basis for a continuous development of this promising and flexible technology.
Picture 4: Integration of SPEEDSIM, and ISSOP for optimal production scheduling

**Literature:**

Power Plant Simulation over the Web
Prof Carl-Johan Fogelholm, Endat Oy
MSc Antti Komulainen, Endat Oy
Dr. Jonnes Kraut Simm-serv
Carl-johan.fogelholm@endat.fi
http://www.endat.fi

Abstract

Information technology development has increased the possibilities of using new tools at power plants. While the computing capacities of common PCs are sufficient for complex plant models and for complicated calculation needs, new tools for applications and ideas have created new ways of implementing simulation tools for process industries and power plants.

Demands for the minimization of impacts on the environment have grown. New software tools help to run plants more efficiently, thus decreasing overall emissions. The use of sophisticated tools is therefore good for the company image.

In recent years, there has been a reduction in the number of plant operators; this has increased the need for software tools at power plants.

Mänttä Energia is meeting several calculation needs by the installation of WebProsim. It can be used in the optimization of the power plant, as it calculates non-measured process values. The software can be used in operator training and for condition tests for steam turbines.

Overview of the company Mänttä Energia Oy

Mänttä has long traditions in the paper industry. The local paper mill of Metsä Tissue Corporation is a part of the Metsäläitto group, which is one of the biggest forest companies in the Nordic countries.

The power station was a part of the factory until 1992, when it was sold and our company Mänttä Energia Oy (MEO) founded. Today, MEO owns the CHP-plant, the hydropower station and the local electric network.

The shareholders of the company are Vapo, fuel supplier, with 45 %, Metsä Tissue, customer and fuel supplier, with 45 % and Mänttä Kaukolämpö with 10 % of the shares. The company has twenty-three employees and six office staff. The customers are Metsä Tissue (steam and power) and Mänttä Kaukolämpö (district heating).

The total boiler capacity in use is about 250 MW and the turbine capacity 25 MW. The power station and its facilities are quite old. Mänttä Energia has three boilers in all. At present, MEO needs only one of our two biggest boilers at a time. Since the end of 1993, MEO has continuously used our main boiler, which is a retrofitted bubbling-bed boiler with a heating output of 94 MW.

Today, the total heat production is about 420 GWh/a and power production 60 GWh/a.

The main fuels are milled peat and sawing waste. In addition to these, Mänttä Energia burns a little oil and coal, as well as sludge from the wastewater treatment plant and, at the same time, paper wastes from the factory. The consumption of peat is about 125 000 t/a and wood 30 000 t/a, while the total consumption of wastes is 30 000 t/a.

The environment is highly valued by Mänttä Energia. It believes that the improved boiler system is quite capable of meeting the aim of burning wastes as well as the main fuels. WebProsim will help operation personnel to operate the plant more economically and will also reduce emissions as the efficiency of the power plant is increased.

Background of the test case and the problem to be solved

A calculation tool was needed for the power plant. The aim was to find software suitable for operator support. The tool should be capable of optimizing plant operations and running performance tests for the steam turbine plant. The software must be able to calculate process data for unmeasured properties such as boiler efficiencies, and CO₂-emissions too. An essential part of the tool had to be an easy–to-use interface. The interface should be very user friendly so that not-so-experienced operators can use it effortlessly.

The main owner of Mänttä Energia, Metsä Tissue, is a big paper producer with many mills in several countries. There is a need, therefore, for the energy manager at the head office to have tools for optimizing
the operation of all the power plants at different mills. WebProsim makes it possible to simulate the operation of any power plant owned by the Metsä Tissue Company.

The improvement approach

The development of Prosim is based on long industrial experience. The leader of the development team, Prof. Carl-Johan Fogelholm, has more than fifteen years’ experience in different positions with manufacturers, utility companies and engineering companies. Prosim is used as a tool for the design and retrofit of power plants, as well as for the conditioning of monitoring systems (CMS) and acceptance testing of power plants. As the chairman for the Nordic Process integration research program, Prof. Fogelholm has very good insight into the latest research in Northern Europe. The optimization of power plants using Exergy and Pinch methods in Prosim is the result of Nordic co-operation.

Combined heat and power production has always played an important role in the Nordic countries. Therefore both industrial and district heating power plants have been included in Prosim from the beginning. In Prosim simulations, it is possible to use all fuels: coal, oil, natural gas and synthetic gases, all types of bio fuels and industrial and municipal waste. Prosim has been in commercial use since 1985.

Prosim is a modular simulation tool for modeling different thermal power plants. It could be used for calculating the same tasks as WebProsim. Prosim is mainly used in the planning of new power plants and simulation of process changes. WebProsim is more suitable for the simulation of existing power plants. It is user-friendlier too, and can be run over the www.

ProsimCMS is an on-line tool in which the instrumentation system and process simulation are connected together. The ProsimCMS System is larger and heavier than WebProsim as it includes on-line connection, database and trend drawing capabilities. Configuration work for ProsimCMS takes much more time than WebProsim. The overall cost of ProsimCMS is two to four times more than the cost of WebProsim.

Excel-based application could handle the main parts of the needed simulation. The accuracy of this kind of simulation is not usually very good and its user-friendliness is likely to be rather poor. The benefit of this kind of simulation is the cost. This could be a good choice for much simpler calculations.

What was done?

The whole project was implemented in close interaction with clients. The main kinds of interaction were e-mail and material published on web pages dedicated to this project.

In this project, a simulation service was built that can be run on a web server, to which clients could connect using only a standard web browser equipped with java 2 support. This design made it easy to test and develop the application in co-operation with clients and end users. There was no need to install any additional software on clients’ computers. For demonstration purposes, Endat set up an HTTP server to run the simulation application and web pages related to the project. The web pages always contained the latest version of the software, ready for testing and evaluation.

Every time a new version of the application was completed, Endat informed clients of its release by e-mail and asked them to evaluate the application. Once they finished testing, they could send their opinion of the application to Endat, together with their ideas as to how to make it better suit their needs. When replies were received, they were analyzed and, if within project limits, were implemented to improve the next version.

Tools that were used

The project was implemented using a fairly large number of different tools, including three different programming languages and various applications needed for writing and compiling source codes. The program responsible for actual simulation has been programmed with Fortran language using Compaq Visual Fortran 6.6 toolkit. The simulation program is controlled by using an interface that is a Java applet written using Sun Microsystems’ Java 2 Platform, Standard Edition (J2SE). Communication between server and client applications is implemented by using common gateway interface (CGI) technology. Server-side CGI-scripts have been written in Perl script language. Web pages relating to the project, including the applet, are published with an Apache HTTP server equipped with Active Perl to run CGI-scripts. The server machine runs with the Microsoft Windows 2000 Professional operating system. The Java source code and many other text documents have been written with the GNU Emacs text editor.

Project Execution

The project started like most software projects with brainstorming during which all possible ideas were discussed. Then the acceptable ideas were written down to form a detailed project plan. The initial data
collection was the most time-consuming work, since the data had to be extracted from many different files and some of the data existed only in handwritten reports.

The simulation model was created using standard Prosim with its Auto CAD interface. This is a relatively easy job for an experienced process engineer.

A very time consuming face was the testing of the simulation model and the user interface. This work needed somewhat more time, and also more engineering hours, than planned.

Customer's role in the work

In negotiations with the customer, details of calculations were cleared up. PI-schemas were studied. The customer introduced the power plant and the information system at site. Fuel processing equipment was studied as well as boilers and steam turbines.

The customer played a major part in the system testing work, which was carried-out in several phases; in each of these, the customer delivered a report. Needed modifications were made, and the customer performed the next testing phase.

System description

The Prosim simulation tool was configured for the power plant of Mäntän Energia. When simulating the whole power plant, needed inputs are given in the flow sheet in white input fields. If an input value is missing, the field is in red. Before starting any calculations, all the input values are evaluated to minimize human mistakes when writing the input data.

Simulation logic of the process

On the consumer’s input screen, the user gives the process steam consumption. Steam consumers are mainly equipment at the paper plant; for example, paper machines. On the boiler inputs screens the user defines the fuel flow that regulates the total steam output of the boilers. This fuel flow is solved in the simulation.

![Figure 1. Main process schema of the power plant.](Image)

In the process schema, the user gives the live steam mass flows for the turbines and the extraction flow for the main steam turbine. The user lets the steam flow into the auxiliary condenser too, and can give the steam properties for the boiler in the main schema or in the input forms.

The steam consumer data, given by the user, is used when solving the process and when the mass flow of the
regulating fuel flow is calculated. If the input values do not give a proper solution, messages are shown to the user and there are also messages that help the user to find the incorrect input data.

Steam turbine condition calculation

This calculation has two phases. The first calculation simulates how the turbine should work in the given load position. This calculation uses steam mass flow into the turbine and extraction mass flows, back pressure, steam cooler temperature and some other temperatures as input.

The second calculation uses all measured process values for the turbine. As a result, we obtain the isentropic efficiencies, turbine constants, the power for each turbine stage and other technical characteristic values that indicate the condition of the turbine.

Ratios for the output parameters “second calculation/first calculation” tell how well the turbine is working. If the ratio is 1.0, the equipment is working as expected. If the ratio is, for example, 0.9, it says the output parameter is 10% smaller than expected at the given load points.

Sim-Serv’s role in the test case

The project was carried out within the Sim-Serv framework; the following organizational aspects deserve specific mention:

The Sim-Serv team was able to arrange contractual arrangements quickly and effectively; once the work was underway, their contact person maintained a positive presence and helped to ensure that work progressed in accordance with an agreed timetable, with deliverables being achieved on time and in full.

Sim-Serv contributed to the collaboration between software developers and industry. The process industry has now better opportunities to utilize the latest software technology.

The Sim-Serv team has developed a capability for providing detailed technical support and also software facilities; this contributes towards our ability to approach the problem equipped with the best information and software systems to meet the job in
hand. Sim-Serv’s ability to act as the independent “consultant” guiding the project to a successful outcome is important and should be developed further in future operations.

Sim-Serv has contributed to collaboration between software developers and industry. The process industry has now better opportunities to utilize the latest software technology. By acting as an independent coordinator, SIM-SERV has maintained confidence between the customer and supplier, also contributing to the completion of the project on time and in full.

Future developments

Endat believes that similar configurations of WebProsim will be installed in power plants in Finland and Sweden within the next few years. Some development work, however, is needed for the users’ interface. The interface should offer a more user-friendly and faster way to configure and specify the computerized simulation model of the power plant. Perhaps this might be a target for future joint development efforts in the framework of Sim-Serv.
LATE PAPERS
APPLICATION OF SIMULATION AND GRAPHIC ANIMATION TECHNIQUES
TO THE STUDY OF MANUFACTURING PROCESSES

Manuel Sánchez-Carrilero
José Miguel Sánchez-Sola
Mariano Marcos
Dept. Ingeniería Mecánica y Diseño Industrial.
Universidad de Cádiz. Escuela Superior de Ingeniería
C/Chile, s/n, 11002 Cádiz, SPAIN

Miguel Ángel Sebastián
Eva Rubio
Dept. Ingeniería Const. y Fabricación
ETSII, UNED
Apdo. 60249, E-28080, Madrid, SPAIN

KEYWORDS
Manufacturing processes, simulation, graphic animation

ABSTRACT
In this work, two alternative methodologies for simulating manufacturing processes are showed. These methodologies are based on the use of computer animation techniques with user codes implemented in commercial software. Both theoretical and practical utilities of these methods are demonstrated.

INTRODUCTION
Manufacturing processes simulation has been widely applied as a first reference for evaluating their operational efficiency and management in the context of the manufacturing system where they are placed. Design, planning and process procedures can be simulated by using either specific computer languages or general purpose computer languages.

However, there is a lack of simulated classic material processing technologies, which are based on basic disciplines, such as mechanics, thermodinamic or dynamic of fluids. In this sense, complete simulated approximations of metal cutting, casting or plastic deformation processes are not usually considered when the global manufacturing process is simulated. This fact can be caused by the high number of computational necessities for processing and programming them.

Additionally, a deep knowledge of the material processing nature is needed in order to define the behaviour model and to implement the corresponding equations and work conditions (Sánchez et al. 2000; Sánchez-Carrilero et al. 1997b). Moreover, computer programming techniques must be widely dominated in order to create the computer systems utility and control tools.

MOVEMENT CONTROL MODELS: DISCRETISATION

Animation involves the reproduction of the realistic movements of the phenomenon which wants to be studied through simulation (Magnerat and Thalmann 1990) Figures 1 and 2. This is the starting point that has conditioned the planification of the different stages of this work. Thereby, firstly 3D-graphic representation of the objects (materials to be processed, tools, machine tools, etc) was carried out by using computer modelling techniques. The geometry of the tools and materials to be processed is discretised by using a finite element analysis FEA geometry pre-processor and an animation commercial software modeller. This methodology requires that the geometrical models developed can be exchanged easily between the graphic systems employed. This geometrical representation is made for a determined instant and it must be extended through different frames in a time evolution in order to give a movement appearance. In this sense, according to (Donikian and Cozot 1995), the most important movement control models can be classified into three categories: descriptives, generatives and behavioural.

Descriptive models reproduce an effect without taking into account its causes, looking for only the movement appearance, although it can be far of the reality (Sánchez-Carrilero et al. 2003). This kind of models is habitually

Figure 1: Animated simulation of a plasma cutting process
Figure 2: Animated simulation of a horizontal turning process

performed by using keyframe techniques and procedural methods such as inverse kinematic of freeform entities deformation, Figures 1 and 2.

Generative models reproduce an effect with description of some of its causes, looking for a realistic animation by integrating in the model the physical laws which govern it. Some examples of these models are rigid object, deformable object and particles systems models (Van Overveld 1994).

Behavioural models simulate the behaviour of autonomous entities with cooperative capabilities based on the environment perception, communication, decision tasks and action (Reynolds 1987). Moreover, simulation by animation involves problems which cannot be analytically treated. Time and space discretisation, and numerical integration techniques must be applied for solving them, all this on the basis of mechanic of continuous and particle systems.

In the model of Mechanic of Continuous (Terzopoulos and Fleischer 1988), surface and volume states of material are studied through values continuously changing with space and time. Two ways can be highlighted. On the one hand, the application of Lagrange’s formulation, which carries on a differential equations system to be solved habitually by numerical treatment through its discretisation along a net of the material and, after this, an integration applying finite differences method. On the other hand, finite elements method (Lin and Yarn 1996) can be also used by using pre-discretised mechanical energies.

When particles system based techniques are employed (Desbrun and Gascur, 1996), only some mass-points, which reproduces the behaviour of the continuous, are considered.

SIMULATION AND IMPLEMENTATION

The aim of this work is focused to obtain realistic representations of the simulation by animation of manufacturing processes through the integration of material behavioural laws and properties with body movement and deformation classic mechanical laws or technological models. The basis of the methodology proposed is included in Figure 3. Looking at this figure, it can be observed as initially the object form is geometrically modelled and the positions and speeds of all the elements of the system are defined. In that moment, material properties are included, and their evolution is monitored by numerical integration and the time by time movement frame is developed.

Two ways can be followed for implementing the methodology above described:

a) Writing directly the high level language code (e.g. C++), which will implement all the details of the processes mathematical model. So, each animation frame will be created and adequately saved. After this, it will be edited in the adequate software giving rise to the final compatible format, Figure 4 (Sánchez-Carrilero et al. 2003).

b) Using SDK (Software Development Kit) of a commercial animation software for creating the plug-ins that animation requires (Sánchez-Carrilero et al. 1997a; Murdock 2001).

CONCLUSIONS

Manufacturing processes optimisation needs a first simulation stage in order to analyse the performance of the system. On the other hand, simulation processes helps to train in the use of machines.

Direct method of simulation allows controlling totally the process but the complexity of the realistic representation can reduce its possibilities. As an alternative, a method based on the use of commercial animation softwares which allows designing dinamic laws and material properties plug-ins is proposed. These plug-ins are developed by programming special modules in a high level language. Some manufacturing processes have been simulated and the results obtained can be considered as satisfactory.

Figure 3: Scheme of the iterative system for movement generation
ACKNOWLEDGEMENTS

This work has received financial support from the Andalusian Government.

REFERENCES


Figure 4: Simulation of a bar extrusion process
A MULTI-OBJECTIVE DECISION SUPPORT SYSTEM FOR OPTIMIZATION IN ENGINEERING DESIGN

Sixto Rios-Insua, Alfonso Mateos, Antonio Jiménez and Luis C. Rodriguez
Department of Artificial Intelligence, Technical University of Madrid
Campus de Montegancedo s/n, 28660 Madrid, Spain
E-mail: srios@fi.upm.es

KEYWORDS
Decision Analysis, Simulated Annealing, Engineering Design, Multi-objective Optimization.

ABSTRACT
Most of the engineering design problems involve the consideration of several conflicting objectives, in such a way that the selection of the best design must be supported by multi-objective optimization. This paper introduces a Decision Support System based on Decision Analysis methodology to aid decision makers in determining an optimal or satisficing design from a set of configurable designs, obtained by simulation of attribute values and supported by an interactive simulated annealing metaheuristic.

INTRODUCTION
The engineering design context is generally characterized by the consideration of several conflicting and noncomensurable objectives that raises the problem of how to manage them to reach a final decision. This setting leads us to the application of the local search metaheuristic simulated annealing in a multiobjective context for providing an optimal or satisficing solution to the design problem.

Such problem is considered in the Multi-objective Decision Analysis (DA) providing a decision tool that follows the steps of the DA cycle, first considering an objectives hierarchy to initialize the modelling process, to aid decision makers (DM) in determining the relevant objectives for the problem under consideration. Next, utility functions are assessed in the lowest-level objectives of the hierarchy, having then that each design has associated a utility vector. The potential designs will be a collection of configurations where each one specifies an admissible value selection for each attribute under consideration and where simulation on the proposed attribute ranges is considered.

The solution approach for identifying the best design based on the DM's preferences is supported by an interactive simulated annealing metaheuristic.

This paper summarizes the implemented techniques in a Decision Support System (DSS) to conduct this multi-objective optimization in the engineering design scenario.

PROBLEM STRUCTURING AND ALTERNATIVE DESIGNS
The first stage of the DA process is to structure the decision problem, which includes specifying objectives and generating alternatives. This is more an art than a science, since there is no formal theory guiding this step. To deal with this complex situation, the system can be used to construct an objectives hierarchy, where the ultimate objective is to provide the best design. Thus, we start constructing the hierarchy by addressing the objectives that represent general concerns of DMs. There are several benefits to be gained from using a hierarchy to model complex decision-making problems with multiple objectives (Brownlow and Watson 1987). For instance, it helps to ensure that there will be no big gaps (missing objectives) at lower levels, situations where redundancy or double-counting could easily occur can be identified and it provides a basis upon which to develop and appraise screening criteria.

Figure 1 shows a simple example of a possible hierarchy constructed by experts for designing a car, which displays two levels and six lowest-level objectives. Note that, it is possible to add or drop a node at any time if deemed appropriate, due to the system flexibility.

Figure 1: Objectives Hierarchy for the Design of a Car
The main purpose is to obtain the design that maximizes the overall impact affecting the market demand. To evaluate the most preferable design, this objective can be split into three subobjectives, concerning the total cost, speed and comfort caused by any design analyzed, and so on, as it is shown in Figure 1.

DA involves establishing attributes associated with the lowest-level objectives to indicate to what extent they are achieved by the considered designs. For some basic properties related to the set of attributes, see (Keeney and Raiffa 1993).

Table 1 shows the attributes under consideration, which also includes their units and relevant ranges.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Measure (units)</th>
<th>Level</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Capital cost</td>
<td>euros x 10^4</td>
<td>35</td>
<td>10</td>
</tr>
<tr>
<td>Running</td>
<td>euros x 10^2</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>Maintenance</td>
<td>euros x 10</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>Acceleration</td>
<td>Seconds</td>
<td>12</td>
<td>4</td>
</tr>
<tr>
<td>Top speed</td>
<td>Km/h</td>
<td>155</td>
<td>260</td>
</tr>
<tr>
<td>Comfort</td>
<td>subjective</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

The most common and useful way of conducting multiobjective optimization is by a previous articulation of DMs preferences, modelled by means of utility theory, as explained in detail in the next section. The DSS allows for imprecision concerning DM’s preferences, thus being the process of assessing the individual utility functions not very demanding, which is less stressful on DMs.

Thus, the starting point will be to establish a set of n attributes, denoted by X_1,...,X_n, and then, the consequences of each design or alternative decision D^i in Omega, where Omega is the available design set, is a vector

D^i \rightarrow (x_{iq},...,x_{iq}) \quad (1)

Now we address the generation of configurable designs: A configurable design set Omega consists of a collection of configurations, where each one specifies a value within the respective range for the attributes under consideration. A set of constraints must be also specified to avoid infeasible combinations by introducing different conditions for attributes value. For example, it is not reasonable to consider cars with top speed higher than 200 km/h and with an acceleration over 10 s or certain comfort level might require a minimum capital cost.

So, we assume that there is a continuous uniform distribution over each attribute range [x^L, x^R], and we randomly generate values from them to provide configurable designs by considering the feasible combinations, i.e., those which verify the proposed rules. Table 2 shows the 7340 feasible configuration designs obtained by applying configurations constraints over 10,000 randomly generated candidates consist of of five values drawn at random within the respective attribute range.

It is important to observe that this type of configurable designs exhibit some combinatorial aspects specially referred to the important increase in the size of set Omega. For example, with only 6 attributes and 12 possible attribute values for each of them, there are closed to 3 million of possible configurations, what gives an idea of the need to control the size of the simulation.

<table>
<thead>
<tr>
<th>Design</th>
<th>Simulated Values for the six attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>D^f</td>
<td>(21,6,7,10,220,0,8)</td>
</tr>
<tr>
<td>D^g</td>
<td>(14,7,3,9,235,0,6)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>D^{100}</td>
<td>(11,8,5,6,170,0,6)</td>
</tr>
</tbody>
</table>

**ASSESSMENT OF IMPRECISE UTILITY FUNCTIONS**

Expected utility theory can be considered as a leading paradigm for normative decision theory. However, in a multiattribute situation, utility theory calls for the DM to provide all the information describing the decision situation to assess component scalar utility functions u_i, which reflects DM pre-ferences on the possible attribute values (Fishburn 1970; Savage 1954; von Neumann and Morgenstern 1947). This can be far too strict in most practical situations, which could lead to the consideration of imprecise component utility functions (Rios Insua 1994; Weber 1987).

The drawbacks associated with utility function assignment are well known, even though good software is available for this process. Several authors, see, e.g. (Jaffray 1989; McCord and de Neufville 1986), have suggested that elicited utility functions are generally method dependent, and bias and inconsistencies may be generated in the assignment process.

Our system attempts to overcome these objections by combining two slightly modified standard procedures: the fractile method and the extreme gambles method, (Farquhar 1984), which belong to the certainty equivalence (CE) and probability equivalence (PE) categories, respectively. Moreover, it assumes imprecision allowing the DM to provide a range of responses instead of a precise value, as the above methods demand for each probability question faced by the DM. This is less stressful on experts, since they are allowed to provide incomplete preference statements, (Rios et al. 1994; von Nitzsch and Weber 1988). As a result, we get a utility function class rather than a single function for each method. The intersection between the two classes provides the range where preferences elicited by the above methods agree. Should such intersections be empty for an interval, the DM would be inconsistent and his/her preferences should
be elicited. The process finishes when a consistent range is achieved.
Specifically, in the fractile method, (Farquhar 1984; Keeney and Raiffa 1993), the DM is asked to provide certainty equivalence intervals or attribute value intervals \( [x^U_i, x^L_i] \) that he/she considers equivalent to different gambles, whose results are the most and least preferred attribute values \( x^B_i \) and \( x^W_i \) with probabilities \( p' \) and \( 1-p' \), respectively. We take \( p'=.25 \), \( p''=.50 \) and \( p^B=.75 \). The method determines a range in the \( x_i/u(x_i) \)-diagram, which can be represented by the bounding utility functions \( u^L_i(x) \) and \( u^U_i(x) \), where \( L(U) \) means Lower (Upper). Figure 2 shows this utility function class drawn between dotted lines.

![Figure 2: Utility Function Classes based on the Fractile and Extreme Gambles Methods](image)

The system includes a routine implementing a wheel of fortune to output these probability intervals, (French 1986), which shows the probabilistic questions and guides the expert until an interval of indifference probabilities is obtained. A number of additional questions are included as consistency checks.

In the extreme gambles method, (Farquhar 1984; Keeney and Raiffa 1993), the DM has to specify probability intervals. We use the upper bounds of the certainty equivalent intervals provided by the DM in the CE-method as sure amounts. Other points may be used for comparison. Figure 2 also shows the utility function class drawn between solid lines and represented by the bounding utility functions \( u^L_i(x) \) and \( u^U_i(x) \) and with \( L \) and \( U \) as above.

As mentioned above, should the intersection of both ranges be empty for some attribute values, the DM would have provided inconsistent responses and he/she should reassess his/her preferences. Thus, the intersection will be the range for the DM’s utility functions. The system is able to detect possible inconsistencies and suggests what the DM should change to achieve consistency. Thus, given a precise consequence \( x_i \) for attribute \( X_i \), we have a utility interval \( [u^L_i(x_{iq}), u^U_i(x_{iq})] \) rather than a single utility function \( u(x_i) \). This is shown in Figure 2 by the striped area for decreasing utility functions.

We have an imprecise utility vector for each design \( D^q \) due to the imprecision concerning utilities,

\[
\left( [u^L_i(x_{iq}), u^U_i(x_{iq})],..., [u^L_n(x_{nq}), u^U_n(x_{nq})] \right) \tag{2}
\]

Now, the problem is to provide a method for ranking the alternative designs characterized by imprecise utility vectors.

**THE OPTIMIZATION PROBLEM BASED ON APPROXIMATION**

In this modelling context we introduce what could be called imprecise utility efficient set, which plays an important role, because it has a similar property to the well-known efficient or Pareto optimal strategy set: given a design in this set there is no other design in \( \Omega \) that dominates it. However, this set can be difficult to determine in this new context, so intelligent approaches are needed to help the DM to arrive at a final solution, as we shall next see.

Under this scenario, we consider an imprecise utility vector (2) associated to each design \( D^q \in \Omega \) and the problem involves choosing the best design \( D^q \), which we rewrite as

\[
u(D^q) = \nu_q\left( [u^L_i(x_{iq}), u^U_i(x_{iq})],..., [u^L_n(x_{nq}), u^U_n(x_{nq})] \right) \tag{3}
\]

Next, we provide a problem-solving approach to solve this complex vector optimization problem. We introduce a preference relation \( \succ_u \) defined on \( \Omega \) from the imprecise utility vector (3), which leads to a dominance principle defined as: given two designs \( D^q, D^{q'} \in \Omega \),

\[
D^q \succ_u D^{q'} \Leftrightarrow \nu_q \geq \nu_{q'} \forall q
\]

which means that \( D \) dominates to \( D^{q'} \) if and only if \( u^L_{iq} \geq u^U_{iq'} \forall i \), with at least one strict inequality. The relation \( \succ_u \) is a strict partial order on \( \Omega \) and, hence, we state the vector optimization problem as

\[
\max_{D^q \in \Omega} u(D^q)
\]

A natural concept is that \( D^q \in \Omega \) is an imprecise utility efficient vector design, if there is no \( D^{q'} \in \Omega \) such that \( D^{q'} \succ_u D^q \) or, equivalently, \( \nu_{q'} \geq \nu_q \). This design set will be called imprecise utility efficient vector set and denoted by \( E(\Omega, u) \). This leads to the problem “Given \( \Omega \) and \( \succ_u \), find \( E(\Omega, u) \)”. Clearly, if the set \( E(\Omega, u) \) had a single element \( D^q \), it would be the most preferred design for the decision-making problem. However, this is not the case in most real
problems, because $E(\Omega, u)$ may contain a lot of elements and is not totally ordered.
Thus, our problem should be restated as "select a single element from the set $E(\Omega, u)$". One way to solve this decision-making problem, favored by behavioral approaches, will be possible if the DM is able to reveal more information on his/her preferences to provide additional structural assumptions to get a subset of $E(\Omega, u)$.
For this purpose, we provide an interactive method, based on multiobjective simulated annealing, to progressively build an approximation set of $E(\Omega, u)$ in collaboration with the DM, which is valid for any of the above settings and based on vectorial optimization for the imprecise utility vectors.

**INTERACTIVE SIMULATED ANNEALING BASED SEARCH**

The underlying idea of this evaluation method, called interactive simulated annealing based on an approximation, (Jiménez et al. 2002), is as follows: The method is based on set $\Omega$, where each element $D^\Theta$ has associated an imprecise utility vector $u^\Theta$ and we generate an approximaton set $A(\Omega, u)$ ($u$ represents the imprecise utility vectors associated with strategies in $\Omega$) of the set $E(\Omega, u)$.

The method begins with a first iteration providing an initial design $D^\Theta$ drawn at random, and thus the set $A(\Omega, u)$ is initialized containing only $D^\Theta$. In the following iterations, another design $D$ from the neighborhood of the solution in the current iteration is considered, and $D$ is accepted if it is not dominated by the designs currently in the approximation set. In this case, we add $D$ to the set $A(\Omega, u)$ and throw out any solution in $A(\Omega, u)$ dominated by $D$. On the other hand, if $D$ is dominated by any element in $A(\Omega, u)$, we would continue considering $D$ for the next iteration with a given probability, known as acceptance probability. In this way, according to the movement in the iterations through the space, we simultaneously build the set $A(\Omega, u)$.

The acceptance probability used in our context is the multiobjective mixed rule, see (Searfoss 1992), as combination of two rules, the Product rule and the Weak rule,

$$P(u_i, u_j, T, \lambda) =$$

$$\rho \min_{i=1}^{m} \left\{ 1, e^{\frac{1}{\lambda_i} \left( E(u_i, \lambda) - E(u_{i-1}, \lambda) \right)} \right\} +$$

$$(1 - \rho) \min_{i=1}^{m} \left\{ 1, \max_{i=1}^{m} \left( e^{\frac{1}{\lambda_i} \left( E(u_i, \lambda) - E(u_{i-1}, \lambda) \right)} \right) \right\}$$

where $\rho$ is a weighting factor provided by the DM. $T$ represents the temperature parameter and a weighting vector $\lambda$ is needed to define each function. So, let $A(\Omega, u, \lambda)$ be the potentially efficient utility solutions set generated by simulated annealing using the weighting vector $\lambda$. Note that by controlling the weights, we shall be able to increase or decrease the acceptance probability of new solutions, which means selecting a certain set of weights that could lead us towards a particular region formed by the potentially efficient utility solutions.

The procedure for obtaining a good approximation $A(\Omega, u)$ of the set $E(\Omega, u)$, is as follows: Taking a weighting vector set $W$, generated in an extensively diversified way, we get, for each $\lambda$, a list $A(\Omega, u, \lambda)$ that contains the potentially efficient utility solutions in the direction induced by $\lambda$. Finally, we filter the set $\bigcup_{\lambda \in W} A(\Omega, u, \lambda)$ by pairwise comparison to output the non-dominated solutions. This filtering process will be denoted by $\Lambda$, in such a way that $A(\Omega, u) = \Lambda \bigcup_{\lambda \in W} A(\Omega, u, \lambda)$.

On the other hand, a set of minimum satisfaction levels $\epsilon_i$ must be provided by the DM for the different attributes, which represent the bounds within which each one of the utilities of the different components satisfies him/her individually. Through an interactive process, the DM will modify these satisfaction levels according to his/her preferences in response to the information obtained as follows.

Thus, the minimum satisfaction levels for the different attributes are initialized. As we start from the imprecise expected utility vector set, the DM will use these minimal satisfaction levels to provide some bounds within each utility of the different components satisfies him/her individually. The DM can fix a single satisfaction level applicable to all attributes or else provide one for each attribute. Throughout the interactive process, the satisfaction levels are automatically updated according to the most preferred solution in the solution list. Moreover, the DM has the option of adjusting these levels, if he/she deems appropriate.

Next, following the intra-set generation described in (Steuer and Harris 1980), a weighting vector is generated. This weighting vector set should be widely diversified to explore the potentially efficient solution set. The elements of an initial weighting vector set $W^0$ are drawn at random from the set

$$\left\{ \lambda \in \mathbb{R}^m : \lambda_i \in [0, 1], \forall i \text{ and } \sum_{i=1}^{m} \lambda_i = 1 \right\}$$

50% of which come from the uniform distribution and the remaining from the Weibull distribution, keeping both of the Weibull parameters at low values as $b = 1$ and $c = 0.3$, so as to achieve a set with 100 times the number of weights (Steuer and Harris 1980). Moreover, the weighting vectors $W^i$ with all components equal to 0 but the $j$th position equal to 1 are also included in this set. Thus, an initial list $L_0 = \Lambda \bigcup_{\lambda \in W^0} A(\Omega, u, \lambda)$ is generated as a result of the filtering operation considered above.

Next, in each iteration, the list output in the previous one, $L_{m-1}$, is presented to the DM, who modifies the minimum satisfaction levels according to his/her preferences on this list. Then, a new restricted list of weighting vectors $W^m$ is generated removing the weights
that become useless with respect to the new satisfaction levels and where a new weighting vector is also introduced so that an empty set of weights is not output (when the DM’s satisfaction levels are too strong) (Jiménez et al. 2002).

Now, we perform multiobjective simulated annealing with the weighting vectors in $W^M$ in order to output the new list of solutions

$$L_M = \bigcap \{L_{M,i} \cup \{A(\Omega, u, \lambda), \lambda \in W^M\}\} \cap \{S_t : u_{M_i}^P \geq \varepsilon_t, \forall t\}$$

(superscript $P$ means average) Note that the minimum satisfaction levels are also used to limit the input of solutions in $L_M$. The interactive simulated annealing ends when the DM is completely satisfied with a solution or set of solutions of $L_M$.

A satisficing design obtained by means of the above method was $D = \{18, 6, 4, 8, 200, 0.7\}$.

CONCLUSIONS

We have developed a DSS for the decision analysis process in engineering systems based on imprecise inputs concerning individual utilities and supported by an objectives hierarchy, which is intended to provide the grounds for the description and evaluation of hypothetical alternative designs for different configurable designs generated by simulation. Relevant objectives as well as attributes associated with the lowest-level objectives of the hierarchy are created in the DSS. The designs must be analyzed with respect to the attributes associated with the lowest-level objectives of the hierarchy. Next, the DM will have to assess the imprecise utility functions, which will be used to provide the inputs for the corresponding model, associating each design with a vector utility. The application of interactive multiobjective simulated annealing leads to the determination of the best or, at least, a small enough set of satisfactory designs, from which the DM could choose the final one more easily by changing satisfaction levels. This means that the user can gain additional insight into the ranking of the designs based on this interactive tool.

ACKNOWLEDGEMENT

This paper was supported by the Spanish Ministry of Science and Technology project DPI 2001-3731.

REFERENCES


Design and Change Management for Mass Customization

Privatdozentin Dr.-Ing. Iris Graessler
Robert Bosch GmbH
Germany
E-mail: iris.graessler@bosch.com

KEYWORDS
Mass Customization, Design Management, Design Methodology, Change Management

ABSTRACT

As mechanics, electrics, electronics and software follow a synthesis trend to mechatronics, customer specific variation can increasingly be offered at competitive prices. This competition strategy known as “Mass Customization” is implemented using a methodology of designing a construction kit for customer specific solutions. A modular product architecture forms the logical backbone of this construction kit. Deduced variants and applications are individualized by selection of standardized, discretely and continuously varying components and cross-disciplinary variation mechanisms. The design methodology is characterized by a systematic, structured procedure based on classic German design theories and takes the specific requirements of mechatronic systems into account. In order to open up potentials of customer specific construction kits, a change process in values and behaviour must take place within each key discipline of company. Following the iceberg model of organizational learning, fields of action within change process as well as success factors of change management are pointed out.

CONTINUOUS IMPROVEMENT PROCESS

The Continuous Improvement Process (CIP) at Robert Bosch GmbH was implemented 13 years ago, initially focusing on production processes. First major results were the broad involvement of employees in optimizing working conditions and increasing output. Within 1½ years, 70% of all employees had already taken part in a workshop aiming at improving their working environment. Further, communication across hierarchy levels was intensified considerably. Instead of reporting from one hierarchy level to another, soon it was taken for granted that employees responsible for a change project themselves present their results to the top management. Gradually a CIP Network combining corporate with decentralized elements was established at Robert Bosch GmbH.

Today, Corporate CIP Coordination ensures harmonization and synergy effects across divisions while a Division’s CIP Coordination takes responsibility for change activities and initiatives in the respective division. The Division’s CIP Coordination works together with CIP Coordinators at each location. Within this CIP network know how, experience and best practice is exchanged and shared. Present main change projects in the division Automotive Electronics (AE) are design management, quality improvement offensive and Bosch Production System.

COMPETITION STRATEGY
“MASS CUSTOMIZATION”

The term “Mass Customization” combines the two contrasting approaches of Mass Production and Customization (figure 1). Mass Production implies cost reduction due to scale effects and gained production experience. Customization focuses on exact fulfillment of customer’s requirements and results in an unique competitive position. Mass Customization therefore aims at producing products to meet individual customer’s needs with mass production efficiency (Tseng, M. et al., 1997). Thus customized products are offered at prices comparable to standard products and continuous individual

![Diagram](image-url)  
Figure 1: Definition of Mass Customization

For producing companies, the focus of Mass Customization lies on individualizing material core products. Often, tailored services related to the core products are offered in addition. Prerequisites of economic success of Mass Customization are mature markets and flexible technologies. Mature markets are characterized by heterogeneous, rapidly changing customer requirements, which can hardly be predicted. Flexible product technologies, such as adaptable materials or mechatronic systems, allow easy adaptation to the individual customer’s preferences. Furthermore, generative or laser driven production technologies make economic production possible in spite of varying characteristics and low lot sizes.

**POTENTIALS OF MECHATRONIC SYSTEMS**

Mechatronic systems emerge from functional shift and extension of mechanics to electrics, electronics and software. As a result of closely interacting disciplines, adaptive and intelligent systems are formed (figure 2). Due to functional integration of mechanics, electrics, electronics and software, the borderline between standard and variable system functions can be moved into areas of low efforts. Software thus advances to a variety driver within mechatronics and increasingly depends on application of specific knowledge. Therefore, mechatronic systems are the most promising application field of Mass Customization for producing companies.

**SOLUTION APPROACH “CUSTOMER SPECIFIC CONSTRUCTION KIT“**

In order to realize a wide range of variation, development must focus on order neutral creation of construction kits for customer specific solutions. With the term “customer specific construction kit” a construction kit is described, from which a defined range of customized products is deduced (figure 3). In contrast to conventional construction kits, the customer specific construction kit is based on a prospective and revolutionary development approach. Prospective means that the construction kit is developed in order to meet future requirements of new products and synergies shall be opened up from the first deduced product on. Revolutionary adds the challenge of a concept which does not rely on already processed customer orders, but provides a foresighted development frame for potential future customer orders. A common product architecture of all deduced products serves as logical backbone of the customer specific construction kit. Deduced variants and applications are individualized by selection of standardized, discretely and continuously varying components and cross-disciplinary variation mechanisms.

Defining validity of the customer specific construction kit forms the basis of effecting a compromise between cost depression and individualization. Besides aspired lot size, the following four dimensions of validity have to be fixed. The range of individualization (1st dimension) characterizes built-in variety. It reflects the spectrum of selection alternatives as well as limiting combination rules and exclusions.

![Figure 2: Potentials of mechatronic systems](image)

![Figure 3: Customer specific construction kit](image)
The defined range of individualization decisively influences how many application development projects can be served by the same construction kit. As counterpole to individualization, the 2nd dimension “level of product hierarchy” stands for standardization. The product hierarchy level of standardization indicates, whether standardized components can be found on level of parts, subassemblies or entire platforms. From production point of view, the number of preferred production technologies and the flexibility of production method are represented by the 3rd dimension “range of production”. With the 4th dimension “temporal stability” intended economic life-time, questions of generation planning, upward- and downward compatibility are addressed.

**DESIGN METHODOLOGY FOR CUSTOMER SPECIFIC CONSTRUCTION KITS**

In order to put such a customer specific construction kit into industrial praxis, a corresponding design methodology has been developed. The subsequently presented design methodology supports a systematic, methodical procedure of order neutral creation of construction kits for customer specific solutions with regard to specifics of mechatronic systems. The methodology meets the following three demands. First, classic German design theories are integrated. Thus, the methodology is based on proven, universal design knowledge and a systematic, methodical procedure is ensured. Second, known methods and tools of creating standards and discretely varying components are used and supplemented by new approaches of designing continuously varying components. For each design phase, a selection of appropriate solution approaches is provided in a clear and well structured manner. Third, cross-disciplinary variation mechanisms are created by integrating and coordinating involved disciplines. According to the respective design phase, needs of coordinating partial solutions between involved disciplines are changing. Therefore, focus of design methodology lies on defining appropriate interfaces between partial solutions in order to form a balanced overall solution. In the following, these characteristics are described using examples.

**PHASES OF DESIGN METHODOLOGY**

Due to underlying prospective and revolutionary development approach, the design task is characterized by a high degree of innovation which results in the need of early design phases, e.g. “establishing function structures” or “finding working principles”. As reference, VDI guideline 2210 E is taken, in which a multitude of German design approaches were unified (VDI 2210, compare also VDI 2206). Based on this phase model, the design methodology for customer specific construction kits is structured into six phases (figure 4).

During the initial phase “planning customer specific requirements” clear limits between standards and variation are drawn within specification. This separation of product characteristics into “standards”, “discretely varying” and “continuously varying” is kept up during entire subsequent design process. Based on specifications, a functional product architecture is derived (phase 2). Partial functions and functional structures are partitioned according to provided variability and involved disciplines. In phase 3, appropriate working principles are selected and cross-disciplinarily connected. Special attention is paid on mutual interactions between chosen principles, effects and algorithms.

Physical product architecture derived from the principle solution makes up the logical backbone of the customer specific construction kit (phase 4). On condition of this common structure, standardization and individualization are balanced. Standard, discretely varying and continuously varying components are designed to be mainly independent from each other and to be recombined. Due to standardized interfaces, the construction kit can be expanded order neutrally as well as order specifically. In phase 5, modular structures and components are detailed. Functionality and compatibility of connected variation mechanisms are checked. Finally, results worked out in phases 1-5 are comprehensively documented in phase 6. Besides product documentation, procedures and rules of handling the construction kit are defined.

**METHODS USE AND RESULTS**

Adequate methods and tools are assigned to each phase of the design methodology for customer specific construction kits. On the one hand, they are structured applying the viewpoints “entire mechatronic system”, “mechanics/electrics/electronics” and “software”. On the other hand, they are separated according to their application into “standards”, “discretely varying” and “continuously varying”. Examples of methods and tools are Quality Function Deployment, generalized elements of Product Line Approach known from software development, morphological boxes, modularization, up-/downscaling, architecture evaluation and compatibility checks. Provision with adequate methods is illustrated using phase 2 “partitioning functional product architecture” as an example (figure 5). Functional product architecture is a commonly used architecture of system functions, whose partial functions are adapted to individual preferences by variation and adaptation In the overview on

---

**Figure 4: Phases of design methodology (Gräßler, I. 2004)**
left side of figure 5, appropriate methods for standards, discretely and continuously varying function structures are offered. Further, the field of application of each method is described. In the given example, the methods “enhanced functional subdividing” and “enhanced functional structuring” are recommended and explained (origins in Beitz, W., 1972, Pahl, G., 1972).

Using enhanced functional subdividing, the entire function is subdivided into partial functions, until these can be separated into “standard (S)”, “discretely varying (V)” and “continuously varying (I)” partial functions. As a guideline, system variety shall be isolated in distinguished partial solutions. Thus, an embodiment structure is prepared from early on, in which standard components are kept distinct from individual components. Partitioning partial functions into standard, discretely varying and continuously varying depends on the disciplines in which the functions shall be realized. Therefore, enhanced functional subdividing and enhanced functional structuring are mutually interacting with each other. Following enhanced functional structuring, partial functions are structured into the corresponding disciplines. In the process, the general guideline is applied, that high degrees of individualization are to be implemented in software. Mechanical, electric, electronic subsystems shall primarily be used to realize standards or discretely varying partial functions. Supplementary, individualization options shall be separated in one discipline only. Thus coordination efforts within development and testing are minimized.

**CROSS DISCIPLINARY VARIATION MECHANISMS**

Besides general tolerance of cross disciplinary variation mechanisms, compatibility of interconnected variety ranges has to be ensured. Basis of coordinating partial solutions between involved disciplines is established by breaking down entire function into partial functions (phase 2, figure 5). Mutual compatibility of discipline bound working principles, effects and algorithms is checked and ensured in phase 3. Only working principles which are compatible among each other, are selected and connected. Due to cross linked disciplines, working principles must not only be compatible within each discipline, but also cross disciplinary. Principle solutions are only valid, if variety ranges of partial working principles complement one another to the functionally required overall variation span. Interactions between alternative states of interconnected working principles are checked, whether they weaken, exclude or intensify each other. Critical constellations must be replaced by alternative working principles. In the context of physical product architecture (phase 4), logical variation possibilities are embodied in terms of components and coordination is optimized as a whole. In this step, cost effects of assigning variety to disciplines become evident. The strategy of cross disciplinary variation mechanisms is put into action by the following approaches. As already known from conventional construction kits, customers select components out of a collection of alternatives as well as additional components can be mounted. As characteristics of construction kits for customer specific solutions these mechanisms are completed by adaptation of discretely varying components and configuration of continuously varying components. Prerequisites are standardized interfaces, independent components and limited interactions between all kinds of components. In phase 5, connected variation mechanisms are investigated on level of the entire mechatronic system and optimized if necessary. In detail, combination of discrete and continuous variation ranges must correspond to the overall variation span defined in the specification. Variation mechanisms determined along development are documented using parameter tables (for discretely varying components) and technical restrictions or constraints (for continuously varying components) in phase 6. Besides direct and desired interdependencies, also unwanted interactions of multiple parameter variation are documented.

---

**Table:**

<table>
<thead>
<tr>
<th>VDI 2210E</th>
<th>establishing function structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>standards</td>
<td>discretely varying</td>
</tr>
<tr>
<td>enhanced functional subdivision</td>
<td>function product variation matrix</td>
</tr>
</tbody>
</table>

**Diagram:**

- **Enhanced functional subdividing**
- **Enhanced functional structuring**

**Key:**
- Information flow
- Energy flow
- Material flow

*Figure 5: Example of methods use in phase 2 (Gräffler, I. 2004)*
CHANGE MANAGEMENT

Successfully initializing, implementing and keeping up a customer specific construction kit, relies on a profound change of values and behaviour in all classical key functions of an enterprise. Special emphasis must be put on sales and marketing, engineering, production and logistics. Therefore, the change process to Mass Customization must be initiated by top management and driven by business units being responsible for realization (top down process). Change process is structured into the phases sensibilization (unfreeze), change und stabilization (refreeze, Schein, E.H. 1997). In analogy to iceberg model from psychology, four steps of organizational learning are passed (figure 6).

In figure 7, these four steps of organizational learning are mapped to the six phases of change processes. Goal orientated change processes have their origin in an overall change mission and are deduced from the strategic orientation of the company. Therefore, the first phase of a change process complies analyzing the present situation. Using internal and external benchmarks, the need for necessary changes is identified. During the second phase change is initiated by planning and establishing priorities. Efforts and benefits are estimated. As a result, the rough organizational structure of the change initiative or project is determined and resources are assigned. When conceiving change in the third phase, the organizational structure is specified in detail and a controlling system indicating the change progress is defined. Sponsors and multiplicators for the change issues must be won. In the fourth phase, project structure is put into action in order to initiate, realize and roll out improvement activities. The project team receives its improvement task from top management. Project leader and change process coach form the head of the change project. In regular reviews, the change progress is reported to a steering committee consisting of members of top and middle management. Additional participants are involved in team meetings or workshops if required. In the fifth phase of change process, the created new organizational behaviour is internalized. Changed procedures and organizational structures are implemented into daily’s work to be run on their own. Thus the ongoing sixth phase of performing learned behaviour in line operation is started.

Fields of action, which are pictured in between phases of change processes and steps of organizational learning in figure 7, are decisive for success of change management. Core element of the field of action “management” is the top management’s decision to adjust selected business units according to the principles of Mass Customization. With the help of “communication” convincing and identification of concerned employees is initiated and supported. Focus of change management lies on the field of action “change project management”. Prior to change process, an overview of numerous aspects to be considered is gained. Goals, expected results and preconditions of change project are contracted between top management and project leader. Goal fulfillment is ensured by profound change project planning and controlling. The field of action “providing methods and trainings” focuses on enabling managers and employees for change process. In the course of “organizational learning and adapting methods” learning processes within organization are triggered and used methods are optimized according to gained experience of first application cases.

CONCLUSIONS AND FUTURE PERSPECTIVES

With the presented design methodology for customer specific construction kits, producing companies are enabled to open up potentials of mass customizing mechatronic systems. In order to put Mass Customization into action, much thought and effort must be spend on change management. However, economic success is not only influenced by decisive new development of customer specific construction kits, but also on their strict application within order processing. As prerequisites, organizational rules of handling construction kits including release processes must be defined and consequently applied. Further, price offers must be calculated according to expenses caused by variety. Last but not least, individualization information must be grasped and processed within order specific deduction from the customer specific construction kit (Grässler, I., 2002).
Figure 7: Fields of action of change management (Gräßler, I. 2004)

REFERENCES
Davis, S. 1987. Future perfect. Addison-Wesley, Reading, Massachusetts
VDI-Richtlinie 2210. Entwurf: Datenverarbeitung in der Konstruktion. VDI Düsseldorf 1975
VDI-Richtlinie 2206 Entwurf. Entwicklungsmethodik für mechatronische Systeme, VDI, Düsseldorf, März 2003

BIOGRAPHY
Privatdozentin Dr.-Ing. IRIS GRAESSLER, born 1969, finished her studies in Mechanical Engineering at Aachen University of Technology in 1995. During 4 years at the Chair for Production Engineering at Aachen University of Technology, she was a project leader of several change projects in the automobile industry. She finished her PhD thesis on topic of Simultaneous Engineering in 1999. She joined the Corporate CIP Coordination of Robert Bosch GmbH in 2000, where she soon became responsible for the Internal Consulting Group “Product Development Processes”. Since December 2002 she runs the CIP Coordination Department in the Bosch division “Automotive Electronics”. In July 2003 she qualified as university lecturer at Aachen University of Technology (Privatdozentin). This article summarizes parts of her recently published book „Kundenindividuelle Massenproduktion“ (Mass Customization).
CAPTURING SHAPE AND COMPLEXITY OF TAILORED APPAREL WITHIN A 3D CAD FRAMEWORK

Marzia Fontana
Dipartimento di Ingegneria Industriale
Università di Parma
Parco Area delle Scienze, 181/A
43100 - Parma, Italy
E-mail: marzia.fontana@kaemart.it

Umberto Cugini
Rossella Mana
Dipartimento di Ingegneria Meccanica
Politecnico di Milano
Via La Masa, 34
20156 - Milano, Italy
E-mail: rossella.mana@polimi.it
umberto.cugini@polimi.it

Caterina Rizzi
Dipartimento di Ingegneria
Università degli Studi di Bergamo
Via Marconi, 5
24044 - Dalmine (BG), Italy
E-mail: crizzi@unibg.it

KEYWORDS
CAD, apparel design, cloth models, physics-based simulation, virtual prototyping.

ABSTRACT
Garment fashion is not only the consequence of styling, but also the result of an effort of design and testing, which involves different and integrated phases of the transformation cycle. In this context, the need to have modeling tools, which enable to design garments directly in a 3D virtual environment is becoming more and more important. In this paper we focus the attention on geometrical and construction issues that should be considered to make possible a complete description and simulation of a real complex-shaped garment within a CAD clothing environment. The analysis on main tailored garment’s design and manufacturing aspects is preceded by an overview on commercial and academic CAD tools to virtually design and simulate garments as well main phases of clothing production cycle.

1. INTRODUCTION
Clothing appears in a large variety of shapes and textile materials, as the result of several design, manufacturing and finishing operations. It arises from the effort of stylists and tailors that choose carefully fibres, textures and colours, profiles of cloth pieces, defining standardized or original rules to combine them together, based on various aesthetic and functional criteria, individual tastes, fashion trends, and production constraints.

Human factors such as creativity of stylists and technical skills of dressmakers for made-to-measure clothing still continue to play a fundamental role, making difficult a complete automation of the whole apparel design process. However, despite this inherent dependency on human skills affecting apparel design, clothing companies increasingly demand fully-integrated apparel-oriented 3D CAD environments. These should support designers not only in the early 2D drafting phase, but also for free-form modelling of garment shapes directly in 3D space, more similarly to the original shape idea as it is conceived in stylist’s mind.

A further need is providing these systems with 3D virtual simulation capabilities, to evaluate/analyse the quality of garment models before production. If such a virtual apparel analysis becomes possible at an early stage of product design, or is made available on-line to customers, in fact, this should allow to reduce the number of garment prototypes, address design specifications closer to final users’ requirements and, definitively, save time and cost. To provide a CAD (and also a web-commerce) system with virtual simulation capabilities, requires a proper underlying physical representation of cloth, describing environmental factors and intrinsic material properties in the form of dynamic/mechanical laws and constitutive material relationships. Recent research on physics-based cloth modelling has proved that physical simulation can, nowadays, guarantee impressively accurate cloth shape predictions; nevertheless, these models analyse simple-shaped one-layered cloth geometries, such as tablecloth, cloth strips, or connected surfaces emulating basic garment samples (tunics, skirts, shirts, etc.) but not actual apparel for real manufacture.

As a result of collaborations with Italian and European clothing companies, we here focus on geometrical and construction aspects that should be considered to make possible a complete description of a real complex-shaped garment within a CAD environment.

As a particularly significant test case, we analyse the complex case of a real-sized laboured man’s jacket considering the several levels of design complexities that have to be considered while defining shape, sewing/assembly rules and aesthetic/functional details of tailored garments.

The analysis on main tailored garment’s geometrical and design features is preceded by an outline on current CAD tools oriented to garment design, considering possible shape definition approaches 2D-onto-3D and 3D-onto-2D. Then, we present the work flow of clothing industry and the critical aspects and the results for the virtual simulation of man’s jacket.

2. STATE OF THE ART OF CLOTHING SYSTEMS
In the clothing sector, differently from other industrial contexts, the extent of automation and the use of computer-assisted systems can still be considered at an early stage. However, a strong demand comes from clothing and fabric furniture industries in order to make available CAD tools to assist the whole cloth design process. Until now, several systems for virtual cloth
Systems for garment design focus on definition/ construction of detailed garment shapes for real manufacture, according to different 2D/3D modelling tools. Software products for cloth visualization aim at producing images that look real for computer animation applications. Finally, hybrid systems, in a certain way, integrate features of both previous categories.

Further on, clothing design tool can be subdivided into two classes: 2D and 3D modelling systems. Since the early ‘80s, several 2D CAD solutions have been proposed on the market specialized in editing patterns, making markers, grading, nesting and printing functions, together with CAM systems for automatically driving cutting and sewing processes. Some examples of full-featured CAD 2D systems are: AccuMark from Gerber Technology (www.gerbercnc.com), Modaris from Lectra (www.lectra.com/en/index.html), TEXWIND from FKGroup (www.fkgroup.com) TUKAcad from TUKAtech (www.tukatech.com/tukacad.htm), PolyPattern from Bontex (www.bontex.it) and PDS from Optitex (www.optitex.com/index.html).

However, clothing companies complained about the lack of garment-oriented packages to design directly in 3D. Initially, CAD prototypes based on garment definition in 3D space have been proposed mainly by research institutes, such as CAD prototypes developed at Queen’s University of Belfast (Hinds and McCartney 1990) Narcissus Apparel CAD developed at Seoul University (http://www.e-dap.net/grizzly/narcissus/english/), 3D Garment System package, developed at Hong Kong University of Science and Technology (Leung et Al. 1997) (Wang et Al. 2003) and the system developed by T. Bonte’s team at the University of Valenciennes (Bonte et Al. 2001) (http://www.univ-valenciennes.fr/sp/mascot). Only few companies has started to develop Apparel CAD of this type, such as the 3D Fashion Builder from Investronica Sistemas (www.investronica-sis.es). As previously mentioned during these last years there has been the need of clothing systems able to simulate garment behaviour. Animation/visualisation tools have been developed to provide such functionality, however most of them are mainly oriented to visualization for movies, or virtual catwalks.

Initially, the 3D garment visualisation happened in commercial 3D CAS through the use of specific plug-in. As mentioned before, clothing simulation requires the definition of a physical representation including environmental constraints and material mechanical properties. Most of current available systems are based on the so-called particle based model (Volino and Thalmann, 2000) (House and Breen, 2000).

Examples of academics packages are MiraCloth, developed at MiraLab laboratory, University of Geneve, (Volino and Thalmann, 2000) and Virtual Dressmaker developed at WSI/GRIS, University of Tuebingen (Keckeisen et Al. 2003) (http://www.tuetex.de).

These plug-in are mainly used for cloth visualization in animated movies and video-games.
Moreover, the demand for virtual apparel prototyping is also emerging for apparel e-commerce via Internet. The Real Time Virtual Try On systems are, for example, MIRALab Virtual Try On, developed at MiraLab laboratory, University of Geneve, D. (Protopsaltou et Al, 2002) and, Tutex developed at WSI- GRIS, University of Tuebingen (http://www.tuctex.de) (Keckesien et Al, 2003).
Recently, the trend aims at integrating the 3D garment visualisation with 2D CAD/CAM systems, namely hybrid systems. Industrial packages adopting this approach are: OptiTex 3D Runway from Optitex (www.Optitex.com), Haute Couture 3D from Pad System Technologies (www.padsystem.com), AccuMark V-Stitcher from Gerber Technology (http://www.gerbertechnology.com/ftw/vw/ 03Prods/cad/pattern/VStitcher.htm), Dressing Sim FDK from Investronica Sistemas (www.investronica-sis.es) and Lectra Systems (www.lectra.com/en/index.php).
Three-dimensional apparel academics CAD System provided, for example, by H. Imaoka and H. Okabe, Kyoto University (Okabe et Al. 1992) and Kang and Sung Min Kim, Seol National University (Kang and 2000).

3. GARMENT PRODUCT DEVELOPMENT PROCESS

The clothing industry consists of all the applications that, starting from raw materials such as woven, knitwear and non-woven, bring to the realization of clothing and accessories for apparel.
Summarising the main phases of clothing productive cycle are: design, industrialization, cutting, manufacturing and finishing. All these phases have an influence on the final product, i.e. the shape of a garment depends on the design and manufacturing solutions adopted to realise it. Therefore it is necessary to know how these phases are carried out in order to develop a 3D CAD system for clothing industry that could be effectively used to design garments. In the following we briefly describe these activities.
The design phase consists of the 2D panels drawing, which determines the nature of the product, and the modelling, which establishes the method for cloth’s manufacturing.
Currently, commercial 2D CAD systems are used. It is possible to create new models or change existing ones also with the aid of digitizers that allow the acquisition of the model’s basic points (perimeter, area and direction). During this phase the basic components (geometry of the 2D patterns) composing the garment are defined. Industralisation consists of all the operations such as, grading and placement that precede the garment’s manufacturing. Grading tools, embedded in commercial CAD/CAM solutions, permit to generate the garment 2D panels in all the required size; while, during placement it is possible to arrange, in interactive and automatic manner, new markers or to change those already existing, to reduce as much as possible woven waste. The markers are successively stamped through a plotter or sent to the automatic cutting unit. This phase is characterized from a considerable diffusion of the consolidated CAD/CAM technologies.
The cutting process takes place automatically by means of proper cutting systems, under the control of the operator. Current CAD/CAM solutions are easily connectible or integrated with the cutting unit to facilitate the process automation and data exchange.
During manufacturing, 2D cut patterns are assembled through sewing operations according to rules resulting from the design activities. The sequence and complexity of manufacturing operations depend on the garment nature and shape (volume): more complex operations are necessary for men’s jacket than for a skirt.
For example, the final shape a man’s jacket is obtained by overlapping and assembling layers of different material (e.g., fabric, linen, canvas, etc.) with different types of sewing or other technological solutions (seams vs adhesive). This means that when modelling and simulating the real behaviour of a garment is not sufficient to consider the geometry of the 2D patterns and the mechanical properties of the fabric. It is necessary to include information regarding the composition of structural parts (e.g., front, collar, revere) of the garment and technological operations (e.g. seams) carried out to obtain such a shape.
Once, all 2D patterns have been assembled, finishing phase takes place. The garment is completed with stylistic details and subjected to different treatments, such as starching and ironing, through the use of industrial press machines. Also in this case, the operations carried out affect theemail shape of the garment. For example, consider a jacket: the plaat of the rever or the collar shape is obtained by executing proper ironing and starching operations.
Therefore, to get closer to cloth manufacturers’ needs, it is necessary to develop tools able to deal not only with 2D patterns geometry and with material properties but also with all aspects related to the various garments’ structural parts (e.g., shoulder paddings, collars) and manufacturing and finishing processes that heavily influence the final garment shape. In the following we highlight some critical issues and functionality required to a 3D CAD system for clothing design.

4. SIMULATING A COMPLEX GARMENT: THE MAN’S JACKET

As previously described, to simulate the real garment behaviour, it is necessary to take into account several issues related to the different phases of the garment product development process. This means that an effective 3D CAD system for clothing design should take into account the cloth’s tri(bology nature:
- Geometry and structure, i.e. geometrical description of garments and relationships among the parts;
- Material, i.e. fabric mechanical/physical properties;
- Process, i.e. design, manufacturing and finishing processes.
In addition, all these key issues are strictly interrelated each other.
We consider as an example, a man’s jacket that is a particularly significant test case of a real-sized laboured garment. It originates from a large number of 2D panels corresponding to front, back and side parts, sleeves, collar, lapel, etc., with complex-shaped borders connected with each other by means of various darts and single/multiple seams.
To illustrate the functionality we have used a 3D garment simulator, named SoftWorld (Cugini and Rizzi, 2002). It adopts the particle based model and it has specifically developed for garment design purposes trying to incorporate all mentioned aspects.

4.1 Geometry and structure

The shape of a garment depends on the geometry of the 2D of basic patterns as well as on the structure of the multi-layered parts. In fact, a male jacket is also the result of a multi-layered fabric composition, following well-defined rules, with fabric layers varying in number among the several jacket’s structural parts and made of different heavier or lighter materials (cotton, canvas, linen, horsehair, etc.). Internal smooth linings, reducing friction against the body, give stability to jacket’s shape, together with reinforcements and tassels.

Figure 2 portrays the multi-layered structure of the jacket front panel while the figure 3 shows the same panel with the specification of fabric/material layers and the geometry of some panels constituting the front multi-layered part. Therefore a 3D CAD system with simulation capabilities really effective for design applications should be able to represent both the geometry of the 2D panel and the multi-layered nature within the physical representation of the garment. To face this problem 2D patterns and the physical model can be subdivided into regions, each one corresponding to a structural part of the jacket, e.g., shoulder, rever, and collar with each region characterized by different physical parameters integrating the multi-layer effects.

Figure 4 shows the 2D patterns corresponding to the right front of the jacket and related regions: (A) shoulder, (B) pleating line of the lapel, (C) facing, and (D) bottom of the jacket.

Figure 5 shows the simulations of the jacket carried out with SoftWorld using a physical representation subdivided into regions and under different conditions: a. without linen and shoulder paddings, b. considering the influence of linen and tassels (used in order to avoid the presence of undulations in proximity of the hems), c. with shoulder paddings.

4.2 Material

Final configuration of a garment depends also on the type of fabric that will be used to manufacture it. Current CAD systems include in their garment model the mechanical properties of different fabrics (cotton, woven, silk, etc.) deriving proper parameters from KES (Kawabata Evaluation System) measurements (Kawabata, 1980). However, results obtained in research projects related to the textile-clothing industry (e.g., UE Brite Project Mascot and Italian project PNR Tessile-Abbigliamento 2000) (Bonte et al. 2001) (Cugini and Rizzi, 2002) demonstrated that to be able to simulate properly a garment behaviour from a physical point of view and to validate the design options, it is necessary to include in the
model also mechanical properties of multi-layered structural parts (see Fig. 5) and seams. This requires the acquisition of mechanical properties through measurements on specimens of jacket's structural parts, e.g., fabric+lining.

4.3 Process

As described in §3, the design, manufacturing and finishing processes heavily influence the final product, i.e., the final shape is reached adopting specific design, manufacturing and finishing solutions. During design phase, the peculiar volume and shape consistency characterizing a complex garment such a male jacket is the result of a multi-layered fabric composition, following well-defined rules, with fabric layers varying in number among the several jacket's structural parts and made of different heavier or lighter materials (cotton, canvas, linen, horsehair, etc.). For example, internal smooth linings give stability to jacket’s shape together with reinforcements and tassels (Figg. 3 and 4). Stuffings for shoulders are placed to correct shape proportions along horizontal directions, emphasizing larger shoulder-torso parts typical of male clothing (Fig. 5). Small aesthetic and functional features enrich the structure of the jacket, e.g., buttons, hooks, external and internal pockets.

During manufacturing process, sewings need to be carefully applied with different degrees of tightness/looseness, depending on position, function and shape curvature effects (e.g., roller), by considering possible differences in the length and shape of borders that have to be joint together, with definition of markers and constraint points.

Finally, both during manufacturing and finishing, the sculptured or smooth volume of a garment effect can be controlled/forced by starching, pleating, ironing, and other several mechanical/chemical actions inducing permanent or semi-permanent deformations on jacket’s fabrics.

Let us show some examples related to the collar, the rever and the sleeve.

The collar, with respect to the item, presents a slight displacement caused of the press action so that its circumference leans comfortably on the garment. Mentioned effect is obtained by using a press for permanent ironing.

The rever pleat assigns to the item a three-dimensional effect caused to the different woven stratification in that region. The press action is used to downsize this volume, and assign the permanent position fold.

Finally, the shape of the sleeve follows the natural posture of the body, this effect is caused of the press action on the sleeve internal stitching.

These effects, obtained through the use of a press, could be simulated acting on mechanical parameters characterising the collar and rever regions.

Fig. 6 portrays the shape of the collar region and some simulation results obtained varying mechanical properties of collar region. In Figure 6b, one can note that the collar is too flat, while in Figure 6b-c the behaviour appears more correct and coherent with the real one (a).

Figure 7 shows the shape rever pleat and simulation results obtained acting on mechanical parameters of the pleat.

Summarising, an effective 3D CAD clothing system should permit to:

- manage geometric data not only for 2D patterns profiles and multi-layered parts, but also to position buttons, hooks or special lines such those of collar, lapel pleats and seams (design process);
- assemble and seam 2D patterns defining a garment according as much as possible to the design specs and manufacturing operations (manufacturing process);
- enrich the garment physical model introducing new operators/parameters in order to be able to simulate permanent material deformations (e.g., lapel pleat) obtained with manufacturing and finishing processes, such as starching and ironing (manufacturing and finishing).

5. CONCLUSIONS

Until now, different systems for virtual cloth modelling have been developed by scientific or commercial communities with different points of view and goals. Research on physics-based cloth modelling has proved that physical simulation can provide accurate cloth shape predictions; nevertheless, these models are not really
adequate to support the designer/modeler during whole the product development process. The analysis on main tailored garment’s design and manufacturing aspects permitted us to highlight key issues and identify the functionalities a 3D CAD clothing system should have to be effective for clothing design applications. The results of tests carried out with the end-users using an academic clothing modeller/simulator named SoftWorld, are encouraging and prove the validity of proposed approach. To get more precise simulation results and to make CAD systems more closer to the real needs of manufacturers, it is necessary to extend system’s CAD functionalities in functional and aesthetic cloth design details, e.g. parametric sewing algorithms (puffed sleeves, close-fitting sewings, jacket rollers, etc.), multi-layered fabric modelling (linings, paddings, etc.), and virtual emulation of manufacturing processes inducing fabric deformations (e.g., by ironing, etc.).

REFERENCES


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 100 publications in the fields of CAD, physics-based modelling and simulation, knowledge-based modelling, virtual prototyping, cloth modelling, and virtual prototyping.

ROSELLA MANA is a Ph.D. candidate at the Department of Mechanics, Polytechnic of Milan, Italy. She received her M.S. Degree in Industrial Design and Master in Virtual Prototypes of the Industrial Design at the same university. She is author and co-author of several publications in the area of industrial design, modelling and simulation.

UMBERTO CUGINI is full professor at the Department of Mechanical Engineering, Polytechnic of Milan, Italy. Previously, he was 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 coordinated several national projects in the area of product design, engineering 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.
AUTHOR LISTING
AUTHOR LISTING

Abdelaziz C. ..................282
Abdelaziz E.-O. ..............282
Anglani A. ..................169
Arrones F.S. ..................97
Arshad H. ..................100
Astheimer P. ..................405
Avenas C. ..................256
Azman S. ..................100

Bagheri A. ..................158
Balasz B. ..................349/61/66
Berruet P. ..................289
Bostelmann S. ..............153
Bouskela D. ..................256
Brandao R.M. ..............12
Brunetta L. ..............361
Bruniaux P. ..............472
Bulsari A. ..................465
Burger L. ..................238
Byrne P.J. ..............311

Camacho A.M. ..............25
Campagne J.-P. ..............175
Caramagno A. ..............392
Carse B. ..................265
Caruana E. ..............7
Carvalho Brito A.E.S. ......335
Casella F. ..............233
Castelain J.-M. ..............437
Cepák M. ..................251
Chen H. ..................108
Cincio R. ..................61
Costantino F. ..............371
Coudert T. ..............289
Cugini U. ..................523
Cunha M.E. ..................353

D’Apice C. ..............299
Daru V. ..................115
Davies T.S. ..............265
De Kee D. ..............449
de la Rubia-Cuestas E.J. ..105
Demoulin M. ..............115
Deshmukh A. ..............81
Desvignes N. ..............256
Di Gravio G. ..............371
Diaz M. ..................243
Díaz-Estrella A. ...........97/105
Domingo R. ..............25

Earnshaw R.A. ..............108
Ehren A. ..................304
El Ouafi A. ..............31
Elia A. ..................169
Escobar C. ..................45
Fedele L. ..................81
Feliz Teixeira J.M. ......335
Fernandez V. ..............392
Fernández-Ramos R. ......180/198
Ferraris G. ..............429
Fethi G. ..................282
Fiacco G. ..................81
Fogelholm C.-J. ..........501
Fontana M. ..............523
Freude T. ..................256

Galindo J. ..................45
Garcia I. ..................88
Gargouri E. ..............319
Garrido D. ..............243
Genco A. ..................73
Gharbi N. ..............397
Ghorashi R. ..............40
Gomez-Padilla A. ..........340
González C. ..............25
Gorsich D.J. ..............139/147
Graessler I. ..............517
Grunow M. ..............304
Guerrero-Garcia P. .......17
Gunter D.D. ..............139
Günther H.-O. ..........205/304

Hammadi S. ..............319
Han C.-S. ..............120
Happi te M. ..............437
Harlin A. ..............465
Haron C.H.C. ..............100
Hascoët O. ..............256
Hasgül S. ..............420
Hassan M.N.R. ..............100
Heavey C. ..............311
Hinestroza J.P. ..............449
Hodgson T.J. ..............442

Insua S.R. ..............512
Joualalen M. ..............397
Javoronkov M. ..............282
Jimenez A. ..............512
<table>
<thead>
<tr>
<th>Author Listing</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joines J.A.</td>
<td>442</td>
</tr>
<tr>
<td>Juslin K.</td>
<td>489</td>
</tr>
<tr>
<td>Kacalak W.</td>
<td>35/61/66/349</td>
</tr>
<tr>
<td>Keshavarzi-Roonizi N.</td>
<td>40</td>
</tr>
<tr>
<td>Khoobkar A.</td>
<td>158</td>
</tr>
<tr>
<td>Kim B.</td>
<td>472</td>
</tr>
<tr>
<td>Kindler E.</td>
<td>20/289</td>
</tr>
<tr>
<td>King R.E.</td>
<td>442</td>
</tr>
<tr>
<td>Koelemejier E.B.C.</td>
<td>366</td>
</tr>
<tr>
<td>Koka N.</td>
<td>187</td>
</tr>
<tr>
<td>Komulainen A.</td>
<td>501</td>
</tr>
<tr>
<td>Koncar V.</td>
<td>458/472</td>
</tr>
<tr>
<td>Koubaa S.</td>
<td>319</td>
</tr>
<tr>
<td>Kraut J.</td>
<td>501</td>
</tr>
<tr>
<td>Krauth J.</td>
<td>489/494</td>
</tr>
<tr>
<td>Krivy I.</td>
<td>20</td>
</tr>
<tr>
<td>Królikowski T.</td>
<td>66/349</td>
</tr>
<tr>
<td>Krug W.</td>
<td>494</td>
</tr>
<tr>
<td>Kurec A.</td>
<td>147</td>
</tr>
<tr>
<td>Landaluze J.</td>
<td>53</td>
</tr>
<tr>
<td>Lawrence P.</td>
<td>415</td>
</tr>
<tr>
<td>Lehmann M.</td>
<td>304</td>
</tr>
<tr>
<td>Letherwood M.D.</td>
<td>139</td>
</tr>
<tr>
<td>Leva A.</td>
<td>233</td>
</tr>
<tr>
<td>Lodewijks G.</td>
<td>324/343</td>
</tr>
<tr>
<td>Lohmann W.</td>
<td>379</td>
</tr>
<tr>
<td>Long A.</td>
<td>479</td>
</tr>
<tr>
<td>Lopes J.A.</td>
<td>353</td>
</tr>
<tr>
<td>Mabesa Jr.J.R.</td>
<td>139/147</td>
</tr>
<tr>
<td>Mahayuddin Z.R.</td>
<td>100</td>
</tr>
<tr>
<td>Majewski M.</td>
<td>35</td>
</tr>
<tr>
<td>Makuch S.</td>
<td>61</td>
</tr>
<tr>
<td>Mana R.</td>
<td>523</td>
</tr>
<tr>
<td>Manzo R.</td>
<td>299</td>
</tr>
<tr>
<td>Marcos M.</td>
<td>509</td>
</tr>
<tr>
<td>Marian H.</td>
<td>175</td>
</tr>
<tr>
<td>Marín-Martín J.</td>
<td>180/198</td>
</tr>
<tr>
<td>Marquandt W.</td>
<td>210</td>
</tr>
<tr>
<td>Martin-Canales J.</td>
<td>180/198</td>
</tr>
<tr>
<td>Martínez A.</td>
<td>53</td>
</tr>
<tr>
<td>Martínez R.J.</td>
<td>88</td>
</tr>
<tr>
<td>Maruri L.</td>
<td>53</td>
</tr>
<tr>
<td>Mateos A.</td>
<td>512</td>
</tr>
<tr>
<td>Mebrahtu H.</td>
<td>384</td>
</tr>
<tr>
<td>Meyer P.</td>
<td>195</td>
</tr>
<tr>
<td>Mileham T.</td>
<td>384</td>
</tr>
<tr>
<td>Miller C.J.</td>
<td>405</td>
</tr>
<tr>
<td>Molina-Conde A.I.</td>
<td>180/198</td>
</tr>
<tr>
<td>Montakhab M.</td>
<td>265</td>
</tr>
<tr>
<td>Montoyes J.R.</td>
<td>175</td>
</tr>
<tr>
<td>Morini M.</td>
<td>429</td>
</tr>
<tr>
<td>Nariman-Zadeh N.</td>
<td>158</td>
</tr>
<tr>
<td>Nassehi V.</td>
<td>40</td>
</tr>
<tr>
<td>Nicolás C.F.</td>
<td>53</td>
</tr>
<tr>
<td>Oh S.-K.</td>
<td>120</td>
</tr>
<tr>
<td>Ottjes J.A.</td>
<td>324/366</td>
</tr>
<tr>
<td>Ozkul T.</td>
<td>410</td>
</tr>
<tr>
<td>Öztürk Z.K.</td>
<td>420</td>
</tr>
<tr>
<td>Pacella M.</td>
<td>169</td>
</tr>
<tr>
<td>Papadakis I.S.</td>
<td>330</td>
</tr>
<tr>
<td>Park K.H.</td>
<td>458</td>
</tr>
<tr>
<td>Park T.-J.</td>
<td>120</td>
</tr>
<tr>
<td>Pipe A.G.</td>
<td>265</td>
</tr>
<tr>
<td>Pla-Castells M.</td>
<td>88</td>
</tr>
<tr>
<td>Porta Nova A.M.O.</td>
<td>12</td>
</tr>
<tr>
<td>Póvoa A.P.</td>
<td>353</td>
</tr>
<tr>
<td>Pujana A.</td>
<td>53</td>
</tr>
<tr>
<td>Pullwitt S.</td>
<td>494</td>
</tr>
<tr>
<td>Puolakka A.</td>
<td>465</td>
</tr>
<tr>
<td>Puri P.</td>
<td>449</td>
</tr>
<tr>
<td>Rennie M.</td>
<td>392</td>
</tr>
<tr>
<td>Reuter M.</td>
<td>153</td>
</tr>
<tr>
<td>Riedewald G.</td>
<td>379</td>
</tr>
<tr>
<td>Rijsenbrij J.C.</td>
<td>366</td>
</tr>
<tr>
<td>Ríos-Gómez F.</td>
<td>180/198</td>
</tr>
<tr>
<td>Rizzi C.</td>
<td>523</td>
</tr>
<tr>
<td>Roberts C.</td>
<td>187</td>
</tr>
<tr>
<td>Robitaille F.</td>
<td>479</td>
</tr>
<tr>
<td>Rodriguez L.C.</td>
<td>512</td>
</tr>
<tr>
<td>Romanin-Jacur G.</td>
<td>361</td>
</tr>
<tr>
<td>Romero-Sánchez J.</td>
<td>180/198</td>
</tr>
<tr>
<td>Rubio E.M.</td>
<td>25/509</td>
</tr>
<tr>
<td>Rudd C.</td>
<td>479</td>
</tr>
<tr>
<td>Sánchez-Carrilero M.</td>
<td>509</td>
</tr>
<tr>
<td>Sánchez-Sola J.M.</td>
<td>509</td>
</tr>
<tr>
<td>Santana G.</td>
<td>218</td>
</tr>
<tr>
<td>Santos-Palomo A.</td>
<td>17</td>
</tr>
<tr>
<td>Saw R.</td>
<td>335</td>
</tr>
<tr>
<td>Schopfer G.</td>
<td>210</td>
</tr>
<tr>
<td>Schuck U.</td>
<td>256</td>
</tr>
<tr>
<td>Schulz J.</td>
<td>195</td>
</tr>
<tr>
<td>Author</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Sebastián M.A.</td>
<td>509</td>
</tr>
<tr>
<td>Shamey R.</td>
<td>453</td>
</tr>
<tr>
<td>Sherburn M.</td>
<td>479</td>
</tr>
<tr>
<td>Shukor S.A.</td>
<td>100</td>
</tr>
<tr>
<td>Siavash A.S.</td>
<td>158</td>
</tr>
<tr>
<td>Sireteanu T.</td>
<td>129</td>
</tr>
<tr>
<td>Sklenar J.</td>
<td>7</td>
</tr>
<tr>
<td>Smyczek J.</td>
<td>278</td>
</tr>
<tr>
<td>Snick B.M.</td>
<td>366</td>
</tr>
<tr>
<td>Solf C.</td>
<td>195</td>
</tr>
<tr>
<td>Stammers C.</td>
<td>129</td>
</tr>
<tr>
<td>Sung W.-J.</td>
<td>120</td>
</tr>
<tr>
<td>Szatkiewicz T.</td>
<td>66/349</td>
</tr>
<tr>
<td>Tang W.</td>
<td>294</td>
</tr>
<tr>
<td>Tanguy A.</td>
<td>20</td>
</tr>
<tr>
<td>Tekam S.</td>
<td>115</td>
</tr>
<tr>
<td>Tetzner E.</td>
<td>379</td>
</tr>
<tr>
<td>Thiriez H.</td>
<td>389</td>
</tr>
<tr>
<td>Thomassey S.</td>
<td>437</td>
</tr>
<tr>
<td>Thoney K.A.</td>
<td>442</td>
</tr>
<tr>
<td>Tronci M.</td>
<td>371</td>
</tr>
<tr>
<td>Vagin V.N.</td>
<td>272</td>
</tr>
<tr>
<td>Veeke H.P.M.</td>
<td>343</td>
</tr>
<tr>
<td>Vegh E.</td>
<td>238</td>
</tr>
<tr>
<td>Vermarien L.</td>
<td>175</td>
</tr>
<tr>
<td>Walker R.</td>
<td>384</td>
</tr>
<tr>
<td>Wan T.R.</td>
<td>108</td>
</tr>
<tr>
<td>Wan T.R.</td>
<td>294</td>
</tr>
<tr>
<td>Wieczorek T.</td>
<td>192</td>
</tr>
<tr>
<td>Williams E.</td>
<td>134</td>
</tr>
<tr>
<td>Yang G.</td>
<td>205</td>
</tr>
<tr>
<td>Yang J.</td>
<td>265</td>
</tr>
<tr>
<td>Yeremeiev A.P.</td>
<td>272</td>
</tr>
<tr>
<td>Zajkowski K.</td>
<td>278</td>
</tr>
<tr>
<td>Zhao X.</td>
<td>453</td>
</tr>
<tr>
<td>Zhou Q.</td>
<td>405</td>
</tr>
<tr>
<td>Zohra. B.</td>
<td>282</td>
</tr>
<tr>
<td>Zottolo M.</td>
<td>134</td>
</tr>
</tbody>
</table>