2^{ND} INTERNATIONAL NORTH-AMERICAN SIMULATION TECHNOLOGY CONFERENCE

on

Soft Computing. Simulation and Software Engineering

NASTEC'2009

EDITED BY

Mokhtar Beldjehem

Lotfi A. Zadeh

Ronald Yager

Madan Gupta

And

Hojjat Adeli

AUGUST 26-28, 2009

GEORGIA TECH
ATLANTA, USA

A Publication of EUROSIS-ETI

Printed in Ghent, Belgium

2ND International North-American Simulation Technology Conference

on

Soft Computing. Simulation and Software Engineering

ATLANTA, USA

AUGUST 26-28, 2009

Organized by

ETI

And

Université St.Anne

Sponsored by

EUROSIS

Co-Sponsored by

Georgia Tech

Ghent University

Hosted by

Georgia Tech Global Learning Center

Atlanta, USA

International Programme Committee

International Programme Committee

Burghard B. Rieger, University of Trier, Trier, Germany
Stuart H. Rubin, Space and Naval Warfare Systems Center, USA
Imre J. Rudas, Budapest Technical University, Budapest, Hungary
Houari Sahraoui, University of Montréal, Montréal, Canada
Aziz Salah, UQAM, Montréal, Canada
Johann Schumann, RIACS/NASA Ames, USA
Antoaneta Serguieva, Brunel University, West London, United Kingdom
Pierre Siegel, University of Aix-Marseille I, Marseille, France
Constantinos I. Siettos, National Technical University of Athens, Athens, Greece
Nematollaah Shiri, Concordia University, Montreal, Canada
James F. Smith, III, Naval Research Laboratory, Washington, DC, USA
Roman Slowinski, Poznan' University of Technology, Poznan, Poland
Dutta Sumitra, INSEAD, Fontainebleau, France
M.N.S. Swamy, Concordia University, Montreal, Canada
Hamid R. Tizhoosh, University of Waterloo, Canada
Jose A. B. Tome, INESC-ID, Lisboa University, Lisboa, Portugal
Enric Trillas, European Centre for Soft Computing, Mieres (Asturias), Spain
Edward Tsang, University of Essex, United Kingdom
Hans Vangheluwe, McGill University, Montreal, Canada
Athanasios Vasilakos, University of Western, Macedonia, Greece
Ronald R. Yager, IONA College, New Rochelle, N.Y., USA
Takeshi Yamakawa, Kyushu Institute of Technology, Kyushu, Japan
Mustapha Yassine, National University of Amman, Amman, Jordan
Ting Yu, University of Sydney, Sydney, Australia
Nevin Vunka Jungum, University of Mauritius, Mauritius
Sahnoun Zaidi, Université de Constantine, Algeria

Reality Mining and Surprise Modelling

Danny Van Welden, KBC, Brussels, Belgium

Industrial Simulation

Guodong Shao, NIST, Gaithersburg, USA

Ecological sustainable development and Innovative technologies for a bio-based economy

Philippe Geril, ETI Bvba, Ostend, Belgium

NASTEC 2009

All articles published in these Proceedings have been peer reviewed

EUROSIS-ETI Publications are ISI-Thomson and INSPEC referenced

For permission to publish a complete paper write EUROSIS, c/o Philippe Geril, ETI Executive Director, Greenbridge NV, Wetenschapspark 1, Plassendale 1, B-8400 Ostend Belgium
.

EUROSIS is a Division of ETI Bvba, The European Technology Institute, Torhoutsesteenweg 162, Box 4, B-8400 Ostend, Belgium

Printed in Belgium by Reproduct NV, Ghent, Belgium
Final Cover Design by Grafisch Bedrijf Lammaing, Ostend, Belgium

EUROSIS-ETI Publication

ISBN: 978-90-77381-50-2
EAN: 978-90-77381-50-2

Preface

NASTEC (**N**orth-**A**merican **S**imulation **Te**chnology **C**onference) is a series of conferences initiated by Eurosis after in-depth discussions with Prof. Mokhtar Beldjehem and North-American Simulationists and Granular Soft Computing Researchers and Practitioners, addressing issues regarding the interplays and synergies of Modeling, Simulation, Granular Soft Computing and their application to the *Software industry*. The second **NASTEC'09** is being held on August 26-28, 2009, at the Georgia Tech Global Learning Center, Atlanta, USA. It has attracted Simulationists, Granular Soft Computing researchers and practitioners, attendees from academic, industry and government agencies in an exchange of ideas and shared experiences. It aims to be the feast of Simulationists in North-America. **NASTEC'09** builds on the success of **NASTEC'08,** by broadening its visibility, and scope of applicability.

The intent of the **NASTEC'09** event is to nurture the spirit of cooperation and strive to improve the quality of life in this *global village* through excellence in hybrid Granular Soft Computing Research and Education by Engineering of next-generation intelligent hybrid Granular Soft Computing systems at the service and for the benefits of humankind. On the one hand, Granular Soft Computing as a *hybrid methodology* aspires to serve as a focal point where the latest results in fuzzy logic (FL), Evolutionary Algorithms (EAs), Probabilistic Reasoning (PR), Machine Learning (ML) and Neural Networks (NNs) are fused together in novel ways in order to transcend the intrinsic limitations of a single methodology, in order to develop hybrid adaptive systems that have the ability to learn and improve their behavior through contact with their environments. Such systems are "good" candidates to tackle hard problems of the *Software industry* successfully. On the other hand, computer simulation is being acknowledged as the "*third leg*" of scientific discovery and analysis, along with theory and experimentation. *Simulation technology* aims at building the *Software digital factory*. The fields of Granular Soft Computing, Modeling and simulation in general have made significant progress; part of it is reflected in the present proceedings volume. **NASTEC'09** attempted to bridge the gap and was able to attract top-level and forefront research; the field itself has brought along a number of new developments, unheard of a couple of years ago. The themes to be discussed this year center around novel issues in connection with Granular Soft Computing, Modeling, Simulation, Simulation-based and Data-driven Software Engineering, Web-centric Computing, Modeling, Educator's Track, Virtual Reality, E-learning and Educational Technology. The program consists of 13 high-quality papers.

We are grateful to a number of people without whom we would not have been able to put the program together. They include our local program committee and international program committee, which have done an excellent job: We received

Preface

2.5 reviews per paper on the average. We would also like to thank many external reviewers who have helped "in the background," and who made sure that we stuck with our schedule. We are grateful to the large number of authors who have considered NASTEC as the target for their work, and even though we could not accommodate every submission, we hope that the reviews will be helpful to many people. Last, but not least, we are indebted to the staff of Eurosis, Sainte-Anne's University and Georgia Tech Global Learning Center for making this event a reality.

NASTEC'09 General Conference Chair,
Mokhtar Beldjehem

Honorary Conference Chairs
Lotfi A. Zadeh, Ronald Yager, Madan Gupta, Hojjat Adeli

CONTENTS

CONTENTS

SCIENTIFIC PROGRAMME

SOFT COMPUTING METHODOLOGY

A UNIFIED GRANULAR FUZZY-NEURO FRAMEWORK FOR PREDICTING AND UNDERSTANDING SOFTWARE QUALITY

Mokhtar Beldjehem
Sainte-Anne University
1589 Walnut Street
Halifax, Nova Scotia, B3H 3S1, Canada
E-mail: mokhtar.beldjehem@usainteanne.ca

KEYWORDS
software quality prediction and understanding, possibility theory, fuzzy sequence, if-then fuzzy weighted rules**,** level of stability, hybrid granular fuzzy-neuro possibilistic model, approximation of Min-Max relational equations

ABSTRACT
We propose herein a novel unified framework that uses a developed hybrid *fuzzy-neuro* system in order to evaluate the impact of inheritance aspects on the evolvability of a class library, and to study the relevance of using inheritance as indicator of class interface stability with respect to version change. To this goal, we propose a novel computational granular unified framework that is cognitively motivated for learning if-then fuzzy weighted rules by using a hybrid neuro-fuzzy or fuzzy-neuro possibilistic model appropriately crafted as a means to automatically extract or learn software fuzzy prediction rules from only input-output examples by integrating some useful concepts from the human cognitive processes and adding some interesting granular functionalities. This learning scheme uses an exhaustive search over the fuzzy partitions of involved variables, automatic fuzzy hypotheses generation, formulation and testing, and approximation procedure of Min-Max relational equations. The main idea is to start learning from coarse fuzzy partitions of the involved metrics variables (both input and output) and proceed progressively toward fine-grained partitions until finding the appropriate partitions that fit the data. According to the complexity of the problem at hand, it learns the whole structure of the fuzzy system, i.e. conjointly appropriate fuzzy partitions, appropriate fuzzy rules, their number and their associated membership functions.

INTRODUCTION AND MOTIVATIONS

Designing a software system so that it can easily evolve as the operating environment changes, as user requirements are modified, and as errors (which inevitably occur in large systems) come to light, is a difficult task and it remains an ultimate goal for software engineering. Building software systems that can evolve rapidly and gracefully in response to changing needs is the most important challenge facing the software industry today.
Pressman (Pressman 1997) estimated to 60% the part devoted to *maintenance* in the total effort of a software development project, of which 80% is devoted directly or indirectly to software *evolution* (adaptive and perfective maintenance). It is widely accepted that developing easily *reusable* and *maintainable* software is of great interest to the software industry, however this still remains an endeavors for the managers and software practitioners alike. Current object-oriented (OO) software systems must satisfy non-functional requirements including quality and performance aspects. These, in contrary to functional requirements, are difficult to assess during the test phase of the life cycle development, i.e., that it is no longer enough that a system meet the functional specification; it has to be also adaptable to future changes. In other terms it has to meet non functional requirement such as those of maintainability, evolvability, etc. Quality prediction models could constitute an interesting solution to such a problem by providing some preventive maintenance layer on software at its early stages of its development life cycle.
Granular and/or soft computing models (Zadeh 1994, 1997, 1998; Yager and Zadeh 1994; Pedrycz 2001; Yao 2000; Gupta et al. 2002; Liu et al.. 2002; Beldjehem 2008) combining several paradigms in general and hybrid fuzzy-neuro models (Beldjehem 1993, 1994, 2002, 2004, 2006, 2008; Sinha and Gupta 1999) in particular could effectively contribute to the building of next-generation intelligent reliable software quality prediction models. The benefits from adopting such hybridization is its capacity to account conjointly for both empirical input-output historical data and heuristic domain knowledge that is available from software engineers, design patterns, good practices and body of knowledge of the software engineering field in general. In addition to resolve the boundary problem, such hybrid fuzzy-neuro models are transparent, tolerant and could effectively ensure accuracy, performance and interpretability. The main idea stems from the possibility to use a hybrid fuzzy-neuro system to generate (tune or extract) a knowledge base (KB) or more specifically a rule base (RB) in terms of fuzzy IF-THEN production rules and thus to generate automatically a fuzzy rule-based quality prediction model by supervised learning from I/O examples. Besides integrating non-linearities directly from the learning examples (training set), the additional advantages of such an approach is the inherited property of *value approximation* which is of paramount importance in exhibiting generalizations necessary to process unseen situations (including testing set and validation set).
On one hand, this has led to intensive research for the development of accurate quality prediction models using various paradigms ranging from Bayesian approach, neural networks, clustering algorithms, conventional ID3 and C4.5 inductive approach of Quinlan (Quinlan 1993) to Fuzzy and inductive fuzzy binary trees approaches (Sahraoui et al.

2000,2001; Boukadoum et al. 2001). However to the best of our knowledge, due to (1) the lack of good practices in software quality and the unavailability of accurate historical data in Software Engineering, (2) the incapacity to account conjointly for both historical data and available explicit linguistic knowledge (3) the lack of interpretability, understandability and explanation facilities; most developed "proof of concept" prototypes are of low accuracy and thus fail to fulfill their "raison d'être", i.e. helping the managers and software engineers in practice in regards to decision making for the purposes of rework and/or refactoring. On the other hand due to the Black-box nature of naïve models in the context of software quality estimation, they are of limited interest from the decision making viewpoint; by consequent they should be replaced with intelligent predictive computational models incorporating explicit formal cause-effect relationships expressed by fuzzy IF-THEN rules that are built or rather machine learned automatically.

A NOVEL LEARNING METHODOLOGY

Motivations for our learning methodology

We want to conduct herein a case study in connection with OO class libraries; which have to preserve, as much as possible, the compatibility among versions in order to maintain software *backward compatibility*. A software is stable in the face of a change to requirements if we do not need to modify it at all. We can sensibly talk of software being more or less stable, depending on the level of change required. Thus stability is indeed a fuzzy concept and is a matter of degree. A software is flexible if it can be readily extended to accommodate likely new requirements with only minimal impact on the existing structure. We propose to develop hybrid soft computing tools that allow the prediction of class evolvability through the symptomatic detection of potential instabilities during the design phase of such libraries. This may help avoid later problems. Evolvability might be defined as the ease with which a software system or a component can evolve while preserving its design as much as possible. In the context of the OO paradigm we might restrict the preservation of the design to the preservation of the library interface. Preserving the library interface must be a continuous goal starting from the initial design. Intuitively, a good design must allow the improvement of existing functionalities and the addition of new functionalities while preserving the library interface. This leads to the problems of assessing the goodness of the design from the perspective of evolvability, and of identifying the internal attributes (coupling, cohesion, size and complexity, inheritance, etc.) that could be used as evolvability indicators. Our current focus is on the validation of the hypothesis that the inheritance aspects of an OO class library might constitute good indicators of its capacity to evolve. To this end, we propose to develop hybrid fuzzy-neuro approach for automatic building and assessing evolvability prediction models. More specifically, we will use our developed hybrid fuzzy-neuro model in order to evaluate the impact of inheritance aspects on the evolvability of a class library, and to study the relevance of using inheritance as indicator of class interface stability with respect to version change. It is worth mentioning that the concept of stability itself (or conversely instability) is a matter of degree and indeed is a fuzzy concept. This is due in part to the fact that intuitively; quality is of intrinsic character and it is inherently a qualitative not quantitative concept. As a result, empirical investigations of measurable internal attributes and their relationship to external quality characteristics are a crucial issue for improving the assessment of software product quality. In these context large measures (known as *metrics*) have been proposed in the literature and have been used in predicting the fault-proneness of classes during design, and for predicting the maintenance effort. In particular it has been shown that size and *inheritance metrics* are good indicators for the stability of a framework. We will focus our attention on how inheritance aspects can be good indicators of the interface evolution of an OO class library. More specifically we will investigate the possibility to learn and understand causal relationships between some inheritance metrics, and the stability of OO library interfaces. Moreover, we will propose the interpretation of the results in terms of weighted fuzzy IF-THEN production rules and the relative importance of the variation of the metrics in relation with the stability of the class library. In our modeling we use *fuzzy metrics* that are *linguistic variable* defined over term sets (or label of fuzzy sets) and represented in terms of membership functions (MFs) and/or possibility distributions. When the possible values for a metric variable are symbolic rather numeric, approximations can be represented in terms of a fuzzy set with a corresponding *membership function* (MF). The stability itself too is modeled as a linguistic variable, this allows coping with several levels of stability (or unstability). In our modeling we use Zadeh's *possibility theory* (Zadeh 1978,1978) and more specifically *possibility/necessity measures* which enables us to accurately estimate how much it is possible that a class is stable (or instable), and how much it is necessary that a class is stable (or instable). We believe that our approach will definitely open the door for intelligent next generation quality prediction systems. Besides machine learning the causal relationships between the inheritance metrics and the stability, they allow the detection of the importance or relevance of each metric to stability which is of paramount importance for an empirical approach of studying and understanding software quality aspects and hence providing justification facilities for the metrics validation issues. Thus enables the understanding of relative importance of each inheritance metric and its influence in the (in) stability of class libraries.

On larger scale, the key mechanisms of object technology (encapsulation, inheritance and polymorphism), offer many opportunities for quality improvement. In fact, in practice the use of objects does not, in it self, improve the quality of software. If anything, object technology introduces new opportunities to introduce defects. A prime example is the misuse of inheritance. A single change in a badly designed class hierarchy can wreak havoc throughout a software system by producing unintended side effects on numerous subclasses and countless instances in a running system. However, inheritance allows a local modification to a single class to produce large scale changes in a system without requiring individual changes to every affected object.

Fuzzy Sets and Fuzzy logic (Zadeh 19965, 1971, 1973. 1979) may be considered as a basis for knowledge and meaning representation and is particularly suited for dealing with natural language and software quality issues. We believe that it is the concept of possibility/necessity distributions (Zadeh 1978, 1979), rather than the truth, that will play the primary role in manipulating such knowledge for the perspective of drawing conclusions. *Possibility theory* (Zadeh 1978, 1979; Yager 1986; Dubois & Prade 1988; Olaf 1998) provides a formal framework for representing and dealing with ignorance, and uncertainties prevalent in modeling real world problems in a flexible computerized manner straightforwardly. It allows handling uncertainty in a rather coherent qualitative way. Two measures of uncertainty called *possibility* and *necessity* are associated with a possibility distribution. These measures turn out to be a convenient tool for modeling of uncertainty, which allows for the representation of imprecise pieces of information, gradual properties, flexible constraints (expressing preferences), incomplete state of information or partial states of ignorance. However it is well accepted that crafting manually fuzzy systems to resolve complex large scale real-world problems is a difficult task that is not always obvious for both the designer (the knowledge-engineer) and the domain expert. This is due partly to the *cognitive limits* of the human being (Miller 1956), but also to the difficulty of understanding the intricacies of dimensionality and inherent complexities and peculiarities of large scale real world problems, and in particular when dealing with complex large scale software systems. Not to mention the lack of precision in the human-human interaction and communication that affects significantly the knowledge acquisition process during the tandem knowledge-engineer/domain expert relationship. Furthermore once it is undertaken it is labour-intensive, costly, error prone, time-consuming, and done on a trial-and-error basis in an adhoc manner and hence need to be totally or partly automated. This is known as the knowledge acquisition bottleneck problem or the *Feigenbaum bottleneck* and is a common problem for all AI approaches. Soft computing as an automated knowledge acquisition methodology aims at remedying such a problem.

Various soft computing (SC) techniques have been used to tackle this learning problem from various points of views. However they are based on some idealizing assumptions and no one adopts a holistic approach to resolve such a problem globally, i.e., finding conjointly appropriate fuzzy partitions, fine tuning the membership functions of the labels used in the rules as well as identifying the structure of the fuzzy system (both the required number of rules and rules themselves explicitly) simultaneously. In practice the required number of rules of the system is not known in advance. Indeed learning fuzzy if-then rules is a difficult multi-parameter optimization problem! We have previously devised, developed, formally validated and deployed a granular hybrid fuzzy-neuro system called Fennec (Beldjehem 1993, 1994, 2002, 2004, 2006, 2008) that was successfully applied to a difficult problem of biomedical diagnosis on Proteins/ Biological Inflammatory Syndromes (B.I.S), to image processing and vision engineering as well as to a complex handwriting pattern recognition problem (Beldjehem 2008). Based on our previous work**,** we propose herein an integrated framework to modify the model, accommodate it and extend its ability and scope of applicability for dealing with software quality prediction by integrating some useful concepts from the human cognitive processes and adding some interesting granular functionalities and knowledge of the software domain. In general, software engineering activities are knowledge intensive and software design is a good application area since the knowledge is generally heuristic in nature and software engineers tend to think on terms of rules.

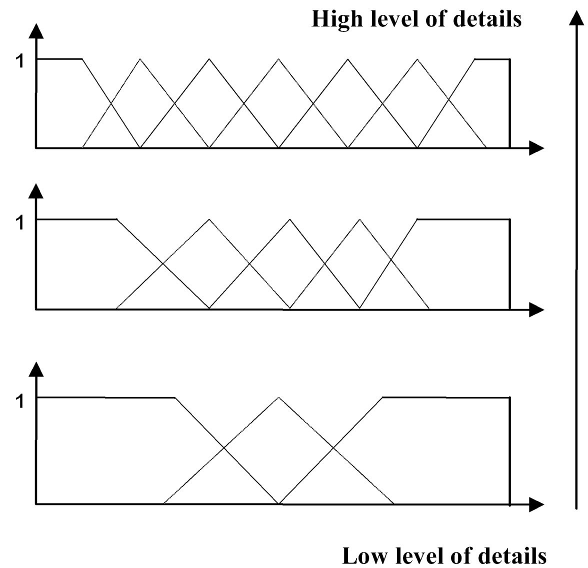


Figure 1 From a coarse fuzzy partition to a fine-grained fuzzy partition

The basic idea underlying our framework stems from the following interesting remarks about human cognition: Let us first focus our attention on the human problem solving process. In solving problems the human starts from a *coarse description* but if needed iterates and goes gradually to a *fine-grained description* or in-depth details enabling more understanding of the underlying problem until reaching a point where one can effectively find a solution and so stops and does not need any more details. At this point, an excess of precision is not needed (is not necessary) because a certain satisfying trade-offs between precision (level of details) and generality of description has been reached and is sufficient and enough for finding a satisfactory approximate solution to the specified problem. Thus after each iteration (increment) a gain of information is obtained enabling more in-depth and more understanding of the underlying situation. Thus, the human converges to a solution gradually by leveraging the level of details. See Figure 1 for more details in connections with a granular soft computing (GrSC) setting. Low levels of details allow coarse or general descriptions reflecting crude approximations whereas high levels of details allow specific descriptions reflecting more or less relatively precise approximations (crisps at the extreme). It is appealing and convenient to mimic mechanically or to emulate computationally such a cognitive process in order to automatically build faithfully by learning an appropriate "*good*" fuzzy prediction system that exhibits both a high accuracy and a good performance for any problem at hand.

This motivates us in building a learning system able to use such abstraction and granulation mechanisms in a fashion that is akin to the way humans achieve problem solving process. In general the required levels of details necessary in describing rules as well as the required number of rules for solving a problem depends to the degree of complexity of the problem at hand and are unknown and hence we propose to detect and determine them by learning within our framework. The rational behind using levels of granularity is obvious for the reader.

THE STATEMENT OF THE LEARNING PROBLEM

Modeling of the software quality prediction problem

It is worth mentioning that the concept of stability itself (or conversely instability) is a matter of degree and indeed is a fuzzy concept. This is due in part to the fact that intuitively; stability is an external characteristic of intrinsic character and it is inherently a qualitative not quantitative concept. Empirical investigations of measurable internal attributes and their relationship to external quality characteristics are a crucial issue for predicting software product quality (Chidamber and Kemere 1994; Li and Henri 1993; Marinescu 2001; Bieman and kang 199;, Henderso-Sellers 1996; Zeuse 1998; Fenton and Neil 1996; Bell 2000; Pigoski 1997, Riel 1996) among others. In these context large measures (known as *metrics*) have been proposed in the literature and have been used in predicting the fault-proneness of classes during design, and for predicting the maintenance effort. More specifically we will investigate the possibility to learn causal relationships between some inheritance metrics, and the stability of OO library interfaces. Moreover, we will propose the interpretation of the results in terms of weighted fuzzy IF-THEN production rules and the relative importance of the variation of the metrics in relation with the stability of the class library. In our modeling we use *fuzzy metrics* that are *linguistic variables* defined over term sets (or labels of fuzzy sets) and represented in terms of membership functions (MFs) or possibility distributions. Referring to figure 2, the (in)stability itself as a concept is modeled as a linguistic variable, this allows coping with several levels of stability (or instability).

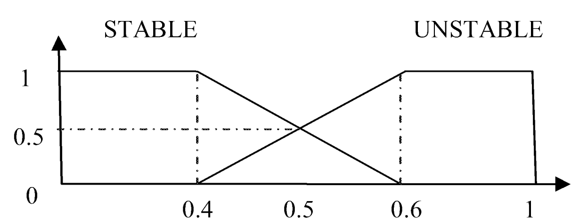


Figure 2 The Fuzzy concept of stability and instability

In our modeling we use Zadeh's *possibility theory* (Zadeh 1978, 1979) and more specifically *possibility/necessity measures* which enables us to accurately estimate how much it is possible that a class is stable (or instable), and how much it is necessary that a class is stable (or instable). We believe that our approach will definitely open the door for intelligent next generation quality prediction systems. Besides learning the causal relationships between the inheritance metrics and the stability, they allow the detection of the importance and/or relevance of each metric to stability which is of paramount importance for an empirical approach of studying for understanding software quality aspects and hence providing justification facilities for the metrics validation issues. Thus enables the understanding of *relative importance* of each inheritance metric and its influence in the (in)stability of class libraries. Ultimately, this enables us to determine the *minimal subset of metrics* allowing to predict the stability (or instability) of library class accurately. We assume herein that we are dealing with the stability issues of either a Java or C++ library or any other object-oriented language. The inputs neurons are herein a certain number of structural metrics (Inheritance metrics of class hierarchy, cohesion, and coupling), our aims is at predicting the stability of class or equivalently to capture the *degree of stability* of the class interface in respects to the evolution of the library from version Vk to an new version Vk+1.

Two output neurons are needed in order to represent the two concepts of stability and instability as illustrated in figure 2.

Conventionally, a software *quality metric* is defined as a function which inputs software data and outputs a single value interpretable as the degree to which software possesses an attribute that affect quality. In departure of conventional methods, we assume herein that a software quality metric is a linguistic variable that might have linguistic values represented by labels of fuzzy sets (such EXTEREMELY SMALL, VERY SMALL, MORE OR LESS SMALL, MEDIUM, MORE OR LESS LARGE, VERY LARGE, EXTEREMELY LARGE) and interpreted by MFs as illustrated in Figure 3. Thus each metric is interpreted by a fuzzy partition or equivalently a *fuzzy sequence*. For the sake of simplicity, we assume dealing with the case of simple inheritance. The inheritance metrics that are assumed to capture the evolution of a class interface are (1) those related to the location of the class in the inheritance tree (2) those related to the ancestors and descendants of the class (3) those connected to the addition, inheritance and overriding of methods, as illustrated in Table 1

Table 1 Class metrics and their descriptions

Metric	Description
DIT	Depth of inheritance Tree. Measure the size of the longest path from a class to a root class within the same inheritance tree.
CLD	Class to leaf Depth. Measures the size of the longest path from a class to a leaf class within the same inheritance tree.
PLP	Position in the longest path. DIT/(CLD+DIT)
NMA	Number of method added New methods in a class
NMI	Number of methods inherited. Methods inherited and not overridden
NMO	Number of methods overridden. Methods overridden
NOM	Number of methods. NMA+NMI+NMO
PMA	Percentage of methods added. NMA/NOM
PMI	Percentage of methods inherited. NMI/NOM
PMO	Percentage of methods overridden. NMO/NOM
NOC	Number of children
NOD	Number of descendants
NOP	Number of parents. NOP $\in$ {0, 1}, which is the case of simple inheritance

Description of the Learning Process

The learning is parametric as well as structural. It has to deal with the complexity of the problem and to discover appropriate knowledge chunks, and approximation heuristics for the problem at hand. Taking into account the degree of complexity of the problem at hand as well as the empirical knowledge contained in the training set, the learning subsystem:

- Identify explicitly the appropriate fuzzy partition for each variable by learning. They are used only as references to generate fuzzy hypotheses. For each variable the appropriate number of granules and the slopes of which will be determined during learning. This information could be either kept or thrown away once the learning is completed without loss of information for the system. As they constitutes only means for generating appropriate membership functions of fuzzy rules and are not used during inference.
- Find the appropriate membership functions for both the antecedents and consequents of every potential rule that is needed to model the problem at hand.
- Ultimately, build the appropriate "good" collection of if-then fuzzy rules (the rule base or knowledge base that consists of a set of linguistic rules), that fits "best" the data that consists of I/O pairs of the training set.

In order to build an automatic workable computational multi-pass learning model some design assumptions are made:

- At each cycle for each input variable X_i the system generates dynamically a fuzzy partition of c granules (starting with c=2, and incrementing c by 1 or 2 at each cycle until reaching a satisfying point). This point constitutes the stopping criterion of our learning mechanism and it reflects too the accuracy level required for the system. It is worth mentioning that increasing c alone does not affect the algorithmic computational complexity of the learning process! It is the number of input variables (n) of the system when it is very large that affects it significantly. We assume to have a reasonable value for n which is almost the case in most classes of real world problems.
- An output variable may be deal with as an input one, but for the sake of simplicity and programmability we assume that a fuzzy partition is given (known a priori for each output variable) and prepared cautiously by the domain expert. As the domain expert is more faced with the difficult problem of capturing relationships between the combinations of inputs variable in relation with a given output variable. In general, for a given output variable the actions (or classes) are well categorized (the number and names of granules are known) by the domain expert even thought the slopes of associated MFs have to be questioned during learning.

FORMULATION OF THE LEARNING PROBLEM

Hypothesis Generation, Formulation and Testing

How to characterize and to represent a fuzzy partition? What operators are needed in manipulating a fuzzy partition? During learning-time, only one operator is needed to create a fuzzy partition having the required known granularity c. It is the repartitioning operator. It consists to divide dynamically during learning-time the universe of discourse into c overlapping granules. It works from scratch, i.e., there is no need for splitting, or fusion or expanding. A partition is used as reference only and its granules do not necessarily constitute MFs for actual rules as they are only used for formulation of initial fuzzy hypotheses during the generation by the systematic exhaustive search algorithm and they are both scale-dependents and context-dependents. We have no other assumption about the fuzzy partition and we are not interested to argue in such matters like "good" partition. The learning will be done at the rule level rather than at the partition level and hence learning a "good" rule is indeed a crucial issue of utmost importance. A fuzzy partition is illustrated in Figure 3 (observe how the rightmost and the leftmost granules are shaped); it is a parameterized family (sequence) of membership functions that cover the universe of discourse for every variable either input or output. It is created dynamically by the execution of the repartitioning operator of granularity equals to c during learning-time. In fact, it is obtained by superposition of two wave functions defined over the same universe of discourse X ranging in the interval $[a_{min}, a_{max}]$. Thus, it is straightforward to extract parameters of granules (MFs) from a given fuzzy partition, as each granule may be considered as an indexed term of the family (or sequence).

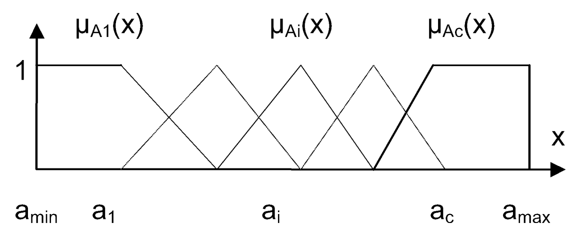


Figure 3 A fuzzy partition of granularity c=5 that is a Superposition of two wave functions representing a fuzzy metric.

Thus, a learning session leverages *graduation* and *granulation* mechanisms conjointly.

A fuzzy partition is represented by vector of c parameters, where c is the granularity level. A computationally more efficient way to characterize it is to use a parametric representation of the MFs of its constituents (called fuzzy members). A fuzzy partition might be thought of as a sequence of granules, each of which is represented by an indexed term. This makes sense as they are computed and manipulated easily like ordinary crisp terms during learning-time. In general as illustrated in Figure 3, every value x of the universe of discourse corresponds to at most two granules. $A_1, A_2 \ldots A_i \ldots A_c$ are just synthetic linguistic labels interpreted by fuzzy sets of normalized MFs. A fuzzy partition might be thought of as a synthetic alphabet that the system create by learning for future hypotheses generation. Thanks to this flexible scale-dependent representation, regardless the range of the universe of discourse of an input variable, the terms of the fuzzy partition sequence are explicitly expressed straightforwardly as follows:

The first term (or granule)

$$\mu_{A1}(x)=\begin{cases}(a_2-x)/(a_2-a_1), & if \quad a_1\le x\le a_2\\ 1 & otherwise\end{cases}$$

For i=2, 3. . . c-1, where c is the granularity of the partition or the i-th term

$$\mu_{Ai}(x)=\begin{cases}(x-a_{i-1})/(a_i-a_{i-1}), & if \quad a_{i-1}\le x\le a_i\\ (a_{i+1}-x)/(a_{i+1}-a_i), & if \quad a_i< x\le a_{i+1}\\ 0 & otherwise\end{cases}$$

And finally the last term

$$\mu_{Ac}(x)=\begin{cases}(x-a_{c-1})/(a_c-a_{c-1}), & if \quad a_{c-1}\le x\le a_c\\ 1 & otherwise\end{cases}$$

Learning by Hybrid Min-Max Fuzzy-Neuro Network

Fuzzy rules attempt to capture the "*rules-of-thumb*" approach generally used by software engineers for decision-making and problem solving. However it is well accepted that crafting manually fuzzy systems to resolve complex large scale real-world problems is a difficult task that is not always obvious for both the designer (the knowledge-engineer, the software designer) and the domain expert. Fuzzy (weighted) rules have been advocated, used, studied and interpreted by many authors (Zadeh 1971; Cayrol et al. 1982; Dubois et al. 1988; Beldjehem 1993; Yager 1996) and machine learned automatically by Beldjehem (Beldjehem 1993). We will focus in dealing with a multi-input single-output (MISO) system as any multiple-input multiple-output (MIMO) system could be converted to a certain number of MISO systems. Let us start with a model overview: As in Beldjehem (Beldjehem 1993) we consider herein to design a fuzzy-neural possibilistic network according to the scheme Fuzzy to Neural (or to switch from fuzzy systems to neural networks). We use fuzzy if-then weighted rules that are herein of the control type instead of the classification type as in (Beldjehem 1993, 1994, 2002, 2004, 2006, 2008) and such a rule looks like:

If (X_1 is w_{k1}, c_{k1}) and (X_2 is w_{k2}, c_{k2}) and (X_3 is w_{k3}, c_{k3}) and

(X_5 is w_{k5}, c_{k6}) **Then** Y_k is V_{ki}

c_{kj} is a weight that represents the grade of importance of " X_j is w_{kj} " in relation with the output Y_k. Thus, conversely the weight a_{kj} =1- c_{kj} represents the grade of unimportance of "Xj is w_{kj}"in relation with the same output Y_k.
Referring to Figure 4, we propose herein a feed-forward fuzzy-neural possibilistic network. We begin with a brief description of the model: two types of weights are associated with the connections.
In fact beyond predicting quality, we are more interested herein by building a class of software prediction tools that justifies and explains its reasoning so that the knowledge and problem solving process is remembered and mimicked by the software engineer in order to tackle the software quality validation and understanding issues. Simply put a system which not only solve the problem of the software prediction but also is able to construct a transparent model for both the software engineer and the software project manager (the human problem solver).

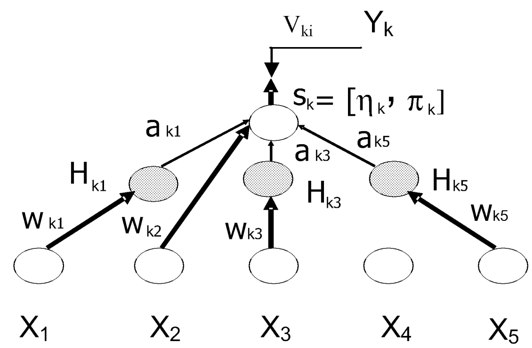


Figure 4 Schematic representation of the hybrid fuzzy-neuro possibilistic Min-Max model used.

Type 1: Direct connections between input cells (X_j) and output cell (s_k) with only synthetic linguistic weights (w_{kj}), interpreted as labels of fuzzy sets, characterizing the variations of the input cells ("X_j is w_{kj} ") with the output cell (s_k), in this case we have a_{kj}= [0,0]=0. Thus ($\prod(X_j;w_{kj}) \vee 0$) = $\prod(X_j;w_{kj})$. Thus the connection between a hidden cell and output cell simply disappears from the graph allowing direct connection.
Type 2: Connections between input cells (X_j) and output cells (s_k) via intermediate cells (H_{kj}), weights associated to connections between input cells (Xj) and intermediate cells (H_{kj}), are herein artificial or synthetic linguistic (w_{kj}), weights associated to connections between intermediate cells (H_{kj}), and output cell (s_k) are herein numerical intervals (a_{kj} $\subseteq$([0,1]), instead of a scalar value ranging in the interval [0,1]) ($a_{kj} \in$ [0,1]).
w_{kj} are unknown artificial or *synthetic linguistic weights* and a_{kj} are unknown confidence interval that reflects a domain of possible values of unimportance for the corresponding connections. Thus providing much more flexibility for the network.
A learning session starts with a "*blank*" fully connected hybrid fuzzy-neuro network without a priori information concerning the weights, i.e. the weights might be thought of as "*placeholders*" only. Learning is parametric as well as structural. Let us consider now cell activation for an arbitrary output cell (s_k), as illustrated in Figure 3, where only connections used in activation of s_k appear. From the semantic point of view, such a figure reflects a neural representation of an if-then fuzzy weighted rule of control type. Let $\prod(X_j; w_{kj})$ = Sup [$w_{kj} \cap X_j$] be possibility measure associated to fuzzy sets w_{kj} and X_j. And let N(X_j; w_{kj}) = Inf [$w_{kj} \cap$ Not X_j] be necessity measure associated to fuzzy sets w_{kj} and X_j . In general our model is governed by the three abstract fuzzy approximate equations as shown below. Thus

allowing the manipulation of fuzzy I/O examples and enabling *approximate learning* reflecting *soft mapping*, this in fact is a departure from conventional learning algorithms.

$$\pi_k = \wedge_{j\in\{1.2.3.5\}} (\Pi(X_j, w_{kj}) \vee a_{kj}) \quad (1)$$

$$\eta_k = \wedge_{j\in\{1.2.3.5\}} (N(X_j, w_{kj}) \vee a_{kj}) \quad (2)$$

$$s_k = [\eta_k, \pi_k] \quad (3)$$

Obviously, each output variable will be assigned an interval as illustrated in equation 3; the inputs of the fuzzy-neuro networks represent the software metrics used in predicting the software class (in)stability and two output variables are required in order to represent the two dual fuzzy concepts of stability and instability respectively in terms of possibility/necessity measures. The interpretation by the means of *linguistic approximations* of the output either stable or instable is as illustrated in Table 2. The process of linguistic approximation consists of finding a label whose meaning is the same or the closest (according to some metric) to the meaning of unlabeled MF (representing either a fuzzy set or an interval) generated by some computational model (learning in our current study).

Table 2 The linguistic approximations of certainty values

Certainty value S_k	Linguistic approximation
[0, 0]	IMPOSSIBLE
[0, 0.05]	ALMOST IMPOSSIBLE
[0, 0.1]	SLIGHTLY IMPOSSIBLE
[0, 0.65]	MODERATELY POSSIBLE
[0, 1]	POSSIBLE
[0.35, 1]	QUITE POSSIBLE
[0.9, 1]	VERY POSSIBLE
[0.95, 1]	ALMOST SURE
[1, 1]	SURE

Observe that Maximum ($\vee$) limits lower amplitudes of inputs, we have $(\Pi(X_j;\ wkj) \vee a_{kj}) = a_{kj}$ if $\Pi(X_j;\ wkj) \leq a_{kj}$, and amplifies higher ones $(\Pi(X_j;\ wkj) \vee a_{kj}) = \Pi(X_j;\ wkj)$, if $\Pi(X_j;\ wkj) \geq a_{kj}$, so the Min-Max composition indicates a somewhat excitatory character. It is worthwhile to notice that Min-Max composition as containing Min and Max operations is strongly nonlinear. Furthermore, such model has been formally validated and it has been shown recently (Beldjehem 2006, 2008) that Min-Max composition preserves the value approximation property. Observe that when $a_{kj} = 1$, the term $\Pi(X_j;\ wkj) \vee a_{kj}$ (respectively $N(X_j;w_{kj}) \vee a_{kj}$) is deleted in the application of Minimum ($\wedge$). Thus ensuring the interpretability and transparency of the model. It is now clear that a_{kj} reflects a notion of unimportance, we point out herein that it is strongly hard if not impossible to make values assignment to grades of unimportance in practical applications, we will propose a mechanism to learn such grades of unimportance. See Table 3, which reflects the metric's effect in relation with the (in)stability of a given class in our framework. In connection with our problem of software quality prediction, semantically, missing edge reflects the non-influence of the input (of the corresponding metric) in the appearance of the output (the stability or instability) of the class. This enables us to determine the minimal subset of metrics allowing to predict the stability (or instability) of library class. Thus enabling the understanding of the class (in) stability issues.

Table 3 The metric's effect in relation with the instability of a given class

The magnitude of the metric's effect	Non-importance
Does not effect	[0.9, 1]
Minimal effect	[0.8, 0.9[
Small effect	[0.6, 0.8[
Medium effect	[0.4, 0.6[
Large effect	[0.1, 0.4[
Very large effect	[0, 0.1[
Partial ignorance	[0, 1]

Thus the fuzzy-neuro possibilistic network might be thought of as a transparent learning device of any non-linear mapping of inputs into an output. It has been proved formally too that Min-Max composition preserves the value approximation property (Beldjehem 2006, 2008) in connections with relational and rule-based fuzzy systems setting.

RESOLUTION OF THE LEARNING PROBLEM

The Learning Algorithm and Implementation Issues

During a learning session the same learning algorithm is used for each output variable Y_j. Let us briefly describe the learning algorithm that is composed of many cycles, each of which is executed as follows: For each output variable Y_j and for each granule belonging to the fuzzy partition that corresponds to Y_j. Iteratively, an initial fuzzy hypothesis corresponds to a combination of certain number of MFs (each of which corresponds to granule of an input variable) is created (formed) by a systematic exhaustive search procedure. Once a fuzzy hypothesis is formed it is loaded or incorporated in the hybrid fuzzy-neuro network weights for test purposes, its components (elements) will be adjusted to fit the training data. Such hypothesis is considered as a potential candidate to be a rule and then is questioned and adjusted during learning by the means of a hybrid fuzzy-neuro possibilistic network using a successive approximation algorithm of systems of Min-Max relational equations. This adjustment is repeated until finding the ones that minimize the signal error. Hence another new combination is then generated and we repeat the same procedure. Thus the obtained adjusted hypotheses that minimize the cost over all possible combinations and that were embedded in the weights of the hybrid fuzzy-neuro possibilistic network are kept in a temporary learning table.

The algorithm proceeds by increasing the granularity and repeats the same cycle, until reaching a satisfying point. In general the learning is stopped when either a certain level of accuracy has been reached or it is impossible or it is computationally worthless to seek minimizing the error much more, i.e. this situation means that increasing the granularity is no more interesting. In general this point constitutes a trade-offs between tractability and low cost solution. Learning need to find an approximate solution that is not necessarily precise (or crisp) optimal one but at the same time it builds a model that do manage to resolve the problem at hand effectively. At the end one or more of the obtained adjusted hypotheses that minimize the cost (over all considered granularity levels) constitutes a valid hypothesis and is transferred and stored in a knowledge base (KB) of the system as it consists effectively of a new learned rule. The system check whether or not a rule is new, i. e. whether or not it is already included the KB, and if necessary, transmits it to the KB, in an intelligible form for the storage (hash table data structure). Assume the system get two or more valid hypotheses, after checking each one, each one is eventually added to the KB as a new rule. The advantage is that by construction (learning) we build a production system with no contradictory rules and thus giving a high satisfactory performance. This is in fact a built-in quality attribute.

Thanks to these granular functionalities, this novel learning algorithm constitutes a departure from the conventional ones, in that it conjointly determine dynamically during the learning-time the required satisfying number of rules necessary to model the problem as well as the rules themselves explicitly. Intuitively, this number is proportional to the degree of complexity of the problem at hand.

The resolution of fuzzy relations equations constitutes a good tool in fuzzy modeling especially for dealing with inverse problems. The *fuzzy relational calculus* theory (Di nola et al. 1989; Beldjehem 1993) provides us with a set of analytic formulas expressing solutions for some types of equations and their systems. However, the existence of solutions of the system is not known in advance. This makes any preliminary analysis rather tedious if not impossible. We reformulate the problem of solving a system of Min-Max from interpolation-like format to approximation-like one. This means that instead of trying to find exact solution, we try to find the best approximate solution. Any scalar and any element of vectors or matrices are assumed to have its value in the interval [0, 1]. Formally; our problem can be stated as follows: "Given an **m x n** matrix **R** and an **n** vector **b**, find an m vector **a** such that ($\mathbf{a}\ \Delta\ \mathbf{R} \supseteq \mathbf{b}$) where Δ is the Min-Max composition and $\supseteq$ denotes the fuzzy inclusion operation. Let us consider the case when there is no solution for the system (it does not satisfy the necessary condition, i.e. $\mathbf{a}\ \Delta\ \mathbf{R} \supset \mathbf{b}$).

This can be also reflected by only computing a distance. Let A, A' be fuzzy subsets of U and α, α' be the corresponding grades of membership vectors. By $\|\alpha-\alpha'\|$ we denote the number $\max_i (|\alpha_i-\alpha'_i|)$, i.e. the maximum of the absolute values of the differences between all element of α and α'. It might be interpreted as the signal error subject to be minimized. Equivalently by using this distance rather than the fuzzy inclusion concept we get the same results; and for this reason we use such a distance $\| \underline{\mathbf{a}}\ \Delta\ \mathrm{R} - \mathbf{b} \|$ in our implementation of the system. It corresponds to minimal distance, hence $\underline{\mathbf{a}}$ is the best approximator. Thus, since our algorithm is valid for both interpolation-like and approximation-like formats, it allows to resolve the more general following problems: "Given an **m x n** matrix **R** and an n vector b, find <u>all</u> m vectors such that $\mathbf{a}\ \Delta\ \mathbf{R} \supseteq \mathbf{b}$". This algorithm is used as approximation procedure by the learning algorithm in our system. The learning consists mainly in crunching (approximating) systems of Min-Max equations while manipulating abstract synthetic linguistic concepts (labels, hypotheses). It can be shown that the best approximator (from the fuzzy inclusion point of view) corresponds to the lower bound $\underline{\mathbf{a}}$ of the inf-semi-lattice. It can be computed straightforwardly using the ε resolution operator only. It has been shown by a worst-case analysis that our computing algorithm has a linear complexity of Θ (m x n) (Beldjehem 1993). In order to illustrate the functioning and the behavior of our *approximation algorithm* let us hand-execute it on the following example, R and b are known. The ε operator is defined as follows (Beldjehem 1993)

$$x\ \varepsilon\ y = \begin{cases} y & \textit{if} \quad x < y \\ 0 & \textit{otherwise} \end{cases}$$

Firstly, we compute the lower bound $\underline{\mathbf{a}}$ of the inf-semi-lattice

$$R = \begin{bmatrix} 0.5 & 0.6 & 0.1 & 0.3 & 0.6 \\ 0.7 & 0 & 0.8 & 0.4 & 0.7 \\ 0.8 & 0.3 & 0.5 & 0.7 & 0.6 \\ 0.4 & 0.8 & 0.6 & 0.8 & 0.7 \\ 0.4 & 0.4 & 0.7 & 1 & 0.6 \\ 0.9 & 1 & 1 & 1 & 0.8 \end{bmatrix}$$

$$b = \begin{bmatrix} 0.3 & 0.3 & 0.5 & 0.4 & 0.5 \end{bmatrix}$$

$\underline{\mathbf{a}} = \vee (\mathrm{R}\ \varepsilon\ \mathrm{b})$, where $\vee$ stands for MAX

$\underline{\mathbf{a}}$ =[0.5 0.3 0 0 0 0]

By performing the Min-Max composition, we have

b =[0.3 <u>0.3</u> <u>0.5</u> <u>0.4</u> 0.5] (the target vector)

$\underline{\mathbf{a}}\ \Delta\ \mathrm{R}$ =[0.4 <u>0.3</u> <u>0.5</u> <u>0.4</u> 0.6]

$\| \underline{\mathbf{a}}\ \Delta\ \mathrm{R} - \mathbf{b} \| = 0.1$

Observe the surprising remarkable approximating power of $\underline{\mathbf{a}}$

Abstract Computational Model of a Learning Session

We are interested herein by establishing the computational abstract model of learning, learning implements a kind of successive approximation of Min-Max system process, and find weights of the hybrid fuzzy-neuro networks that fits "best" the data that consists of pairs I/O of the training set**.** Formally, from the computational point of view, for each output (s_k), a learning session consists to resolve or to approximate (r + 1) systems of Min-Max equations, as follows:

$$a\ \Delta\ R^{(0)} \supseteq b$$

$$a\ \Delta\ R^{(1)} \supseteq b$$

$$\vdots$$

$$a\ \Delta\ R^{(r)} \supseteq b$$

Learning consists to prefer (validate) the configuration (the fuzzy hypothesis) of the best approximate solution (from the fuzzy inclusion point of view), i.e. which minimizes the local cost function and hence the corresponding deep structure. In other terms the learning process finds incrementally the "*best*" deep structure which corresponds to the following matrix $R^{(l)} : l \in [0,r]$ such that:

$$a \Delta R^{(j)} \supseteq a \Delta R^{(l)} \supseteq b, \forall j=0...r$$

Or equivalently,

$$\| a \Delta R^{(j)} - \mathbf{b} \| \geq \| a \Delta R^{(l)} - \mathbf{b} \|, \forall j=0...r$$

Learning tries progressively by *successive approximation* to minimize the local cost function by the generation and the approximation of a new system. Thus, this approximation algorithm constitutes the mathematical machinery of learning. It has been shown that this system is a *universal approximator* (Beldjehem 2006, 2008), furthermore it is now clear that the ultimate aim of learning is to generate a consistent system which correspond to exact solution (or to establish a *universal interpolator*), however it seems that is not always the case in practical applications. In general the value of the local cost function may be seen as a *quality index* for a learning session or a *performance index* for the system. Learning has high speed due to its simplicity and analytic nature. The learning consists mainly in crunching (approximating) systems of Min-Max equations while manipulating abstract synthetic linguistic concepts (labels, hypotheses). Indeed the fuzzy learning process may be thought of as a new kind of algorithmic fuzzy optimization or rather an algorithmic *fuzzy approximation*.

CONCLUDING REMARKS AND FUTUR WORKS

We have developed a cognitively motivated granular computational framework for learning fuzzy systems and have showed how to use it properly and effectively in order to resolve a software quality prediction problem. This allows the automatic learning of fuzzy if-then quality prediction rules of object-oriented software systems which are large scale, too complex or too ill-defined to admit of precise quantitative analysis, description or quality control strategy. It may be thought of as an automatic means or a learning device for capturing the description of ill-defined concepts, relations and decisions rules. Such a framework integrates conjointly both the perceptual and the cognitive aspects of the human problem-solving process and ensure a granular processing of the underlying input from different granularity levels. It is the first attempt in the field. Implementation of a system called Neofennec (that is an upgraded or refined version of Fennec) working under the proposed framework is underway. The "*good*" prediction rule-base (RB) is obtained automatically from I/O training examples. Its inference engine has the inherent ability to generalize, which permit it to classify unseen examples accurately. During learning-time the system finds automatically the adequate levels of details (granularities) for the problem at hand. It is possible using a *linguistic approximation* to build automatically completely a true linguistic fuzzy prediction system by learning.

The other promising alternative that constitutes a candidate solution is to use an *evolutionary algorithms* (EA) as in (Pedrycz 1997; Falkenauer 1998; Cordon et al. 2001), EAs are optimization techniques based on the mechanics of natural selection and natural genetics. EAs has a great power for global optimization and do not need to know the model previously. EAs also do not require the continuity of the parameters. Therefore EAs can easily handle the multi-parameter problems of software quality prediction and for this reason it seems appealing and convenient to use EAs too in our framework. Thus an EA may replace the generator of hypotheses subsystem in our framework. Instead of using an exhaustive search to generate all possible fuzzy hypotheses to test, it may be possible to use an EA that converges to the "*best*" hypothesis by evolution rather than trying all possibilities. EAs can effectively contribute significantly in our framework thanks to their learning and optimization capabilities. In particular to try to fuzzify concepts used by EAs to obtain and use *fuzzy fitness* functions (or *fuzzy cost*), *fuzzy crossover*, *fuzzy mutation* and so on to ensure *smooth evolvability* during learning.

Even though we are more interested in (soft) computation rather than (natural) cognition, i.e. in developing new, powerful and useful tools that learn for resolving software engineering problems such as quality, effort and cost estimation, we believe that as we understand better how to build these computational systems we'll start to have theories that are powerful enough to explain some aspects of the software engineering agenda. We believe that our approach will definitely open the door for intelligent next generation software quality prediction systems. Besides learning the causal relationships between the inheritance metrics and the stability, they allow the detection of the importance and/or relevance of each metric to stability which is of paramount importance for an empirical approach of studying for understanding software quality aspects and hence providing justification facilities for the *metrics validation* issues. Thus enables the understanding of relative importance of each inheritance metric and its influence in the (in) stability of class libraries. Ultimately, this enables us to determine the *minimal subset* of metrics allowing to predict the stability (or instability) of library class and thus implements an effective *software assistant* enabling software designers, programmers and project managers to predict the software quality and to handle, to rework, to refactor, to plan and control the related software development activities especially those connected to "*programming in the large.*" of complex large scale projects.

The second part of this investigation could be to extend our methodology of our empirical study to tackle three similar problems that consist to measure (1) *reusability* (2) class *fault-proneness*, and (3) *cost estimation*. We propose also to extend our investigation to comparing and assessing our approach with related work in terms of accuracy, performance-interpretability tradeoffs and finally to draw conclusions and propose future promising directions for research. We are exploring to use our framework for the purposes of rework and/or refactoring in order to improve the design of existing code as in (Fowler 1999). Anyway it remains that software "units of work" measure a social activity and not production or productivity in the sense of manufacturing.

We believe that hybrid granular soft computing, software engineering, machine learning, knowledge-based systems, simulation, performance evaluation have to learn from each other, and could be integrated or fused synergistically (not competitively) in order to build next generation of intelligent software (quality, effort and cost) prediction systems. Such systems exhibit performance-accuracy trade-off, adaptability, transparency, interpretability, robustness, tractability, tolerance for uncertainty, categorization abilities, value approximation and therefore ensuring *smooth evolvability* and *generalization capacities* which are required in coping with the evolution and maintenance of complex large-scale software systems and dealing with their ever changing requirements and dynamic environments. Soft computing models (SCMs) are becoming major tools for software quality prediction. Ultimately, granular and/or soft computing as a *hybrid methodology* aspires to serve as a focal where the latest results in Fuzzy Logic (FL), Probabilistic Reasoning (PR), EAs, Machine Learning and Neural Networks (NNs) are fuzzed together in novel ways in order to transcend the intrinsic limitations of a single methodology, in order to develop hybrid adaptive systems that have the ability to learn and improve their behavior through contact with their environments, such systems are "good" candidates to tackle a class of hard problems of software engineering successfully.

REFERENCES

Beldjehem M. 1993. "Un apport à la conception des systèmes hybrides neuro-flous par algorithmes d'approximation d'équations de relations floues en MIN-MAX : le système Fennec." Ph.D. Thesis in Computer Science and Software Engineering, Université de la Mediterannée, Aix- Marseille II.

Beldjehem M. 1993. "Fennec, un générateur de systèmes neuro-flous." in Proc. les Actes des Applications des Ensembles Flous, Nimes, France, 209 -218 (in French).

Beldjehem M. 1993. "Le système fennec." in Electronic BUSEFAL 55, 95-104 (in French).

Beldjehem M. 1994. "The fennec system." in Proc. ACM Symposium on Applied Computing (SAC), Track on fuzzy logic in Applications, 126-130, Phoenix, AZ (March).

Beldjehem M. 2002. "Machine Learning based on the possibilistic-neuro hybrid approach: design and implementation." in Electronic BUSEFAL 87, 95-104.

Beldjehem M. 2002. "Learning IF-THEN Fuzzy Weighted Rules." in Proc. International Conference of Computational intelligence, Nicosia, North Cyprus.

Beldjehem M. 2006. "Validation of Hybrid MinMax Fuzzy–Neuro Systems." in Proc. International conference of NAFIPS, Montreal.

Beldjehem M. 2008. "Towards a Validation Theory of Hybrid MinMax Fuzzy–Neuro Systems." in Proc. the WSEAS International Conference, Sofia.

Beldjehem M. 2008. "A Validation Theory of Hybrid MinMax Fuzzy–Neuro Systems." in Proc. the CIMSA International Conference, Istambul.

Beldjehem M. 2008. "A Unified Granular Neuro-Fuzzy Framework for Learning Fuzzy Systems." in Proc. the NASTEC International Conference, Mc Gill, Montreal.

Beldjehem M. 2008. "A Granular Neuro-Fuzzy Framework for Recognition of Arabic Handwriting: The GoavmRec System." in Proc. the NASTEC International Conference, Mc Gill, Montreal.

Beldjehem M. 2008. "Granular Unified Framework for a Machine Visual System." in Proc. the NASTEC International Conference, Mc Gill, Montreal.

Bell D. 2000. Software Engineering: A programming Approach Addisson-Wesly Longmn Publishing Co., Inc., Boston, MA, USA.

Bieman J.; Kang B.-K.1995. "Cohesion and reuse in an Object-Oriented System." ACM Symp. Software Reusability.

Boukadoum M. ; Sahraoui H. A. and Lounis H. 2001. "Machine Learning Approach to Predict Software Evolvability using Fuzzy Binary Trees." in Proc. of the IC-AI'01, Las Vega.s

Cayrol M.; Farreny H. and Prade H. 1982. "Fuzzy Pattern Matching." Kybernetics 11, 103-116.

Chidamber S. and Kemere C. 1994. "A Metric Suite for Object-Oriented Design." IEEE Trans. On Software engineering, 476-492.

Cordon O.; Herrera F.; Hoffman F. and Magdalena L. 2001. Genetic Fuzzy System, Evolutionary Tuning and Learning of Fuzzy Knowledge Bases, World Scientific.

Dinola A.; Sessa S. and Pedrycz W. 1989. Fuzzy Relation Equations and Their Applications To Knowledge Engineering, Kluwer Acdemic Publisherss, Dordrecht, NL.

Dubois, D. and Prade H. 1988. Possibility Theory: An Approach to Computerized Processing of Uncertainty, Plenum Press, New York, USA.

Dubois D.; Prade H. and Testemale C. 1988. "Weighted fuzzy pattern matching", Fuzzy sets and systems 28, 313-331.

Falkenauer. E. 1998, Genetic Algorithms and Grouping Problems, John Willy and Sons.

Fenton N. E. and Neil M. 2000. "Software Metric: Roadmap", in Proc. Of the 2nd International Conference on software engineering.

Fowler M. 1999. Refactoring-Improving the design of existing code, Addison-Wesley, 1st edition.

Gupta M.; Jin L. and Homma N. 2002. Static and Dynamic Neural Networks: From Fundamentals to Advanced Theory, John Wiley and Son Inc., New York.

Henderson-Sellers B. 1996. Object-Oriented Metrics: Measures of Complexity." Prentice-Hall.

Liu T. Y.; Yao Y. Y. and Zadeh L. A. 2002. "Data Mining, Rough Sets and Granular Computing,", Physica-Verlag, Heidelberg.

Li W. and Henry S. 1993. "Object-Oriented Metrics that Predict Maintainability." Journal of Systems and Software. Vol. 23, No. 2

Marinescu R. 2001. "Detecting Design Flaws via Metric in OO Systems." In Proc. of tools IEEE Computer Society.

Miller G. 1956. "The magical number seven, plus or minus two." The Psychological Review 63, 81-97.

Newell A. and Simon, H. 1972. Human Problem Solving, Englewood Cliffs, NJ: Prentice Hall.

Olaf W. 1998. Possibility Theory with Application to Data Analysis, John Wiley & Son Inc.

Pedrycz W. (Ed.). 1997. Fuzzy Evolutionary Computation, Kluwer Academic Publishers

Pedrycz W. (Ed.). 2001. Granular Computing: An emerging paradigm, Springer. Series: Studies in Fuzziness and Soft Computing 70.

Pigoski T. M. 1997. Practical Software Maintenance, Wiley Computer Publishing.

Pressman 1997. Software Engineering, A Practical Approach, 4th edit., Mc Graw-Hill.

Quinlan J. R.1993. C 4.5: Programs for Machine Learning, Morgan Kaufmann Publishers.

Riel A. J. 1996. Object-oriented Design Heuristics, Addison-Wesley.

Sahraoui H. A. and Boukadoum M. 2001. "Extending Software Quality Predictive Models Using Domain Knowledge." in Proc. of the 5th International ECOOP, Budapest.

Sahraoui H. A. ; Boukadoum M. ; Lounis H. and Ethève F. 2000. "Predicting Class Libraries Interface Evolution: an investigation into machine learning approaches." in Proc. of the 7th APSEC'00.

Sinha N. K. and Gupta M. 1999. (Ed.). Soft Computing and Intelligent Systems: Theory and Applications, Academic Press, New York.

Yager R. 1986. Fuzzy Set and Possibility Theory: Recent Developments, Pergamon Press: New York.

Yager, R. and Zadeh, L. A. 1994. Fuzzy Sets, Neural Networks and Soft Computing, Van Nostrand Reinhold: New York.

Yager R. 1996. "On the interpretation of Fuzzy If-Then Rules," Applied Intelligence 6, 141-151.

Yao Y. Y. 2000. Granular computing: basic issues and possible solutions, Proceedings of the 5th Joint Conference on Information Sciences, 186-189.

Zeuse H. 1998. A Framework of Software Measurement. Walter de Gruyter.

Zadeh L. A. 1965. "Fuzzy sets." Info. Control 89, 338-353.

Zadeh L. A. 1971. "Toward a theory of fuzzy systems." in : R. E. Kalman, N. Declaris, Eds., Aspects of Network and System Theory (Holt, Rinehart and Winston, New York), 209-245.

Zadeh L. A. 1973. "Outline of a new approach to the analysis of complex systems and decision processes." IEEE Trans. Syst. Man Cybernet. 3, 28-44.

Zadeh L. A. 1976. "Fuzzy sets and information granulation." in Advances in Fuzzy Set Theory and Applications, in M. Gupta, R. K. Ragade, R. R. Yager (Eds.), North –Holland Publishing Company, 3-18.

Zadeh L. A. 1978. "Fuzzy sets as a basis for a theory of possibility." Fuzzy sets and syst. 1, 3-28.

Zadeh L. A. 1979. "A theory of approximate reasoning." in Machine Intelligence 9 (J.E. Hayes et al.; Eds). Elsevier, 149-194.

Zadeh L. A. 1984. "A theory of commonsense knowledge." in Aspect of Vagueness (H.J. Skala, S. Termini and E. Trillas; Eds). Dodrecht: Reidel, 257-295.

Zadeh L. A. 1994. "Fuzzy logic neural networks, and soft computing." Communications of the ACM 37, 77-84.

Zadeh L. A. 1997. Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, Fuzzy Sets and Systems 19, 111-127.

Zadeh L. A. 1998. "Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information / intelligent systems." Soft Computing 2, 23-25.

Application of Kriging to Predict Wireless Network Connectivity

Soroor Al Khafaji
Heping Liu
Alice E. Smith
Auburn University
3301 Shelby Center
Auburn, AL 36849
E-mail: smithae@auburn.edu

KEYWORDS

Kriging, Prediction Modeling, Wireless Network Connectivity.

ABSTRACT

The research aim of this paper is to investigate the effectiveness of a new Kriging model which uses Taylor expansion to predict wireless network connectivity. Wireless network connectivity is measured by the strength of emitted signal power from the tower to the point in question. The prediction results are compared with those from the literature where an Ordinary Kriging model and a neural network are used to conduct the same prediction. Root mean squared error (RMSE) and maximum absolute relative error (MARE) show that the prediction results of the new Kriging model are much better than those obtained before with average differences from 51.56% to 85%. This study shows the promise of the new Kriging model to accurately estimate wireless signal strength.

1. INTRODUCTION

Wireless network research deals with all types of connections without cables. Network connectivity is defined as the ability to connect to a network connection point with some minimum threshold of signal strength and quality that allows ongoing communication (Nasereddin et al., 2005).
The signal to noise ratio is a measure of network quality and it is a ratio of desired signal to undesired signal (noise) in the average level of transmission. The distance of the point in the network to the transmitting tower serving the point and the maximum value of transmitted power are used to calculate the signal to noise ratio.
This paper uses a new Kriging model based on Taylor expansion (Liu, 2009) to predict the signal to noise ratio, that is, quality of connectivity, for wireless networks of 14-towers, 27-towers, and 45-towers (found in Konak, 2009, Nasereddin et al., 2005 and Bartolacci et al., 2004). The prediction results from Kriging are compared with those of the references. The rest of this paper is as follows. The second section introduces Kriging approach. The third section presents the empirical analysis and compares the prediction. Finally, some conclusions are provided.

2. KRIGING APPROACH

Kriging was originated in geostatistics (Matheron, 1963) and revolutionized that field. Several empirical studies proved its superiority over other interpolating techniques such as splines (Laslett, 1994). Recently, Kriging has been widely used in many fields including cost estimation (Chaveesuk and Smith, 2005; Baioumy, S. M. H. et al., 2008), simulation interpolation (Barton, 1994; Kleijnen and van Beers, 2005), and optimization (Huang et al., 2006; Liu and Smith, 2008). The application of Kriging to wireless wave propagation can be found in Leflbvre et al., (1996) and Yu et al. (2006).
Kriging is a geostatistical interpolation technique similar to IDWA (Inverse-Distance Weighted Average) which estimates the elevations at the reference points. That is, Kriging uses the combination of weights at known points to estimate the value at unknown point (Forkuo, 2008). It fits a mathematical function to a specified number of points, or all points within a specified radius, to determine the output value for each location.
Kriging assumes that the variable $Z(X)$ can be written as the sum of a deterministic component $\mu(X)$ and a stochastic component $R(X)$ (Kumar, 2007):

$$Z(X) = \mu(X) + R(X) \qquad (1)$$

The deterministic component $\mu(X)$ is the expected value of the regionalized variable $Z(X)$ at location X, which is a vector. A fundamental assumption of Kriging is that the covariance between any two locations depends only on the distance between the two locations and can be expressed as a function of the distance. In Simple Kriging, it is assumed that the mean $\mu(X)$ is zero across the field of interest. Ordinary Kriging is the most commonly used type of Kriging. It assumes that $\mu(X)$ is a constant but unknown nonzero mean. In the third type of Kriging, Universal Kriging, the mean is assumed to have a functional dependence on spatial locations and can be approximated by a chosen model with the form below (Liu, 2009):

$$\mu(X) = \sum_{j=1}^{k} \alpha_j f_j(X) \qquad (2)$$

where α_j is the j^{th} coefficient to be estimated from the data, $f_j(X)$ is the j^{th} base function of spatial coordinates that describes the drift of the mean, and k is the number of base functions. The Kriging estimator is given by a linear combination:

$$\hat{Z}(X) = \sum_{i=1}^{n} \lambda_i Z(X_i) \qquad (3)$$

Weights $\lambda_i, i = 1 \dots, n$ are chosen to satisfy the following statistical condition:

$$E\left[\hat{Z}(X) - Z(X)\right] = 0 \qquad (4)$$

$$var\left[\hat{Z}(X) - Z(X)\right] = minimum \quad (5)$$

The Konak approach (Konak, 2009 and Bartolacci et al., 2004) to estimation uses Ordinary Kriging and requires the calculation of a new Kriging model for each point to be estimated, however the Kriging we use herein allows for the calculation of a single model to be used over the entire range of operation.

The Kriging model in this paper uses Taylor series expansion to approximate $\mu(X)$ and can improve the prediction accuracy of Kriging (Liu, 2009). In Taylor Kriging, Taylor expansion is used to identify base functions. Taylor expansion has very good nonlinear functional approximating capabilities and magnitudes, sample standard deviation simplifies the parameter setting of influence distance. thus can assist Kriging to capture the data mean drift. In Taylor Kriging, sample standard deviation is used as the measurement unit of influence distance in the covariance function. Since different problems have different data magnitudes, sample standard deviation simplifies the parameter setting of influence distance.

3. COMPUTATIONAL EXPERIENCE

3.1 Application of Kriging

Data of wireless networks with 14, 27 and 45 towers from Konak, 2009, Nasereddin et al., 2005 and Bartolacci et al., 2004 are used as the test problems. The inputs are the points for which signal strength is to be predicted and the power of the tower serving that point (that is, the closest tower). The output is the log (base 10) of the signal to noise ratio. In (Konak, 2009), the inputs are the *x*, *y* coordinates of the towers and the sets of points in the target coverage area and the transmitted power of the tower serving that point. We also use the latter input, however instead of the former we use the Euclidean distance *d* of point *i* to tower *j* input.

We use 3rd order Taylor expansion in the Taylor Kriging with an influence distance of two standard deviations for the base function. The prediction results from the Taylor Kriging are compared with the actual results and the prediction results of Ordinary Kriging (called Konak Kriging in this paper) and a neural network (NN) also from (Konak, 2009).

Note that the magnitudes of the input parameters are significantly different (distance and power). To avoid the influence of differing data magnitudes on predicted values we normalize the data to *z* values.

We considered three different strategies for training (that is, calculation) of the Kriging model and testing of the calculated model. These are 25% training, 50% training and 75% training. The total data set for each problem was 358, 322 and 288 points for the 14, 27 and 45 tower problems, respectively. So, the 50% training strategy for the largest problem would use 144 data points to calculate the model and 144 data points to test the model. Points were randomly selected by Konak (2009) and we used the same points as in that paper (see Figure 1 for an example).

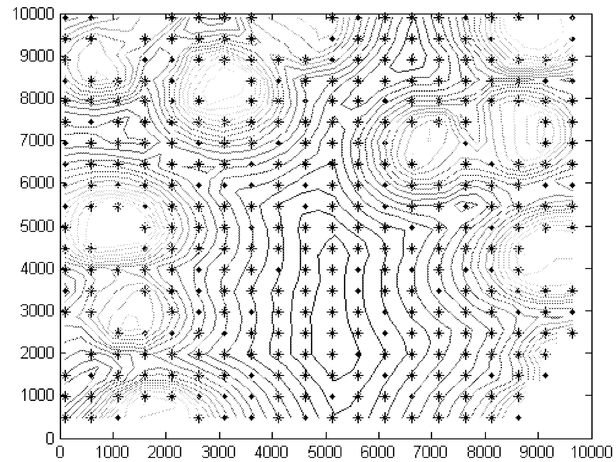


Figure 1: The Coverage Map of the 14-Tower Problem and the Training Set for 75% Partitioning. * Denotes the Points where the Signal-to-Noise Ratios are Sampled, and • Denotes Test Points

The root mean squared error (RMSE) and the maximum absolute relate error (MARE) are the performance measurement standards of the prediction.

3.2 Results Discussion

Table 1 summarizes and compares the results from Taylor Kriging, Konak-Kriging and NN for the 14-Tower, 27-Tower, and 45-Tower networks with different percentages of training and testing points. In this table, the errors from Taylor Kriging are all significantly less than the others approaches except for the shaded parts. For the maximum sized training set for the smaller problems, all methods performed comparably. However when training data is constrained or the problem is large, Taylor Kriging was much better.

	25% Training, 75% Testing	50% Training, Testing	75% Training, 25% Testing
14-Tower Wireless Network			
Konak-Kriging			
RMSE	3.7	1.7	1.3
MARE	89.3	24.6	4.2
NN			
RMSE	3.5	2.1	1.3
MARE	72.2	17.0	3.9
Taylor Kriging			
RMSE	2.1	1.7	1.5
MARE	51.4	14.8	3.3
27-Tower Wireless Network			
Konak-Kriging			
RMSE	3.4	2.7	1.4
MARE	61.0	54.0	285.0
NN			
RMSE	3.8	3.0	2.0
MARE	357.0	50.0	379.0
Taylor Kriging			
RMSE	2.05	2.4	1.4
MARE	11.3	40.0	99.0
45-Tower Wireless Network			
Konak-Kriging			
RMSE	3.5	8.5	2.6
MARE	59.0	411.6	57.9
NN			
RMSE	4.1	3.4	2.9
MARE	51.9	339.5	77.3
Taylor Kriging			
RMSE	2.2	2.1	1.1
MARE	46.6	297.5	19.0

Table 1: 1.Results for Different Numbers of Training versus Testing Points

Table 2 shows the average values of RMSE and MAREs for the three types of training and testing sample sets. The average differences of MAREs between Taylor Kriking and Konak-Kriging and NN are 51.56% and 85%, respectively. The predicted results from the wireless networks with 50% training and testing sample sets are used as examples to diagrammatically describe the prediction performance of the different methods in Figures 2 through 4. The real data (log of signal to noise ratio) is sorted from low to high so that the figures can easily show the prediction accuracy. An entry of less than zero indicates noise greater than signal (a ratio < 1). The figures show that the predictions are unbiased. Predictions for larger problems are less accurate than for smaller problems.

	25% Training Plan	50% Training Plan	75% Training Plan
Konak-Kriging			
RMSE	3.5	4.3	1.8
MARE	69.8	163.0	115.7
NN			
RMSE	3.8	2.8	2.1
MARE	160.4	135.6	153.4
Taylor Kriging			
RMSE	2.1	2.1	1.3
MARE	36.4	117.4	40.5

Table 2. Average Errors across All Problems

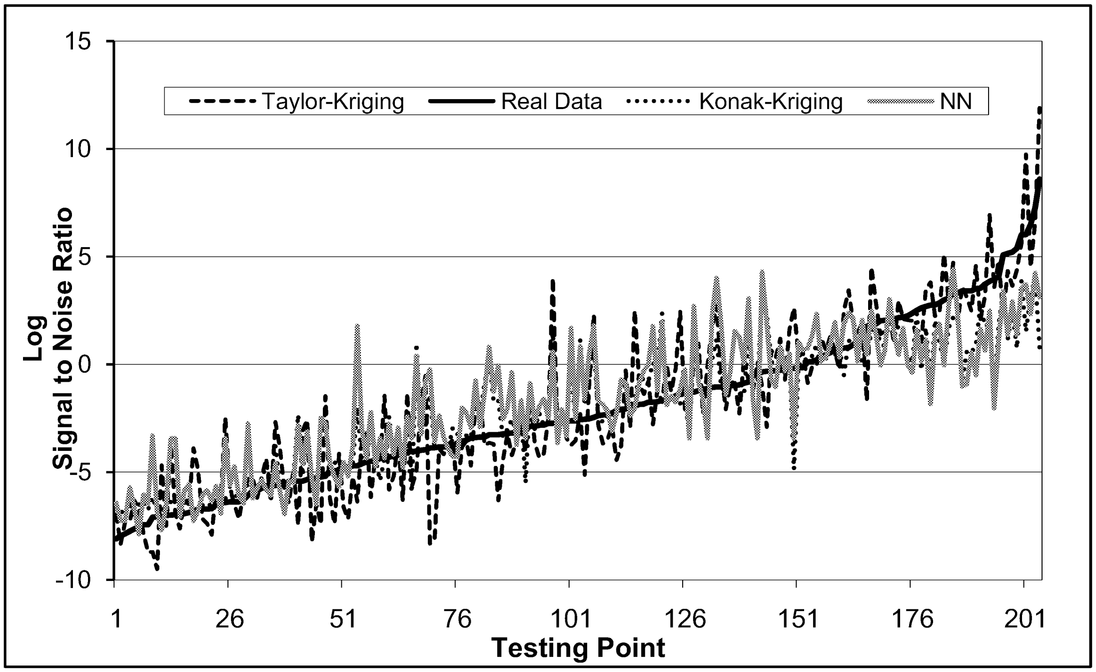


Figure 2: 14-Tower Wireless Network Comparison at 50% Training and 50% Testing

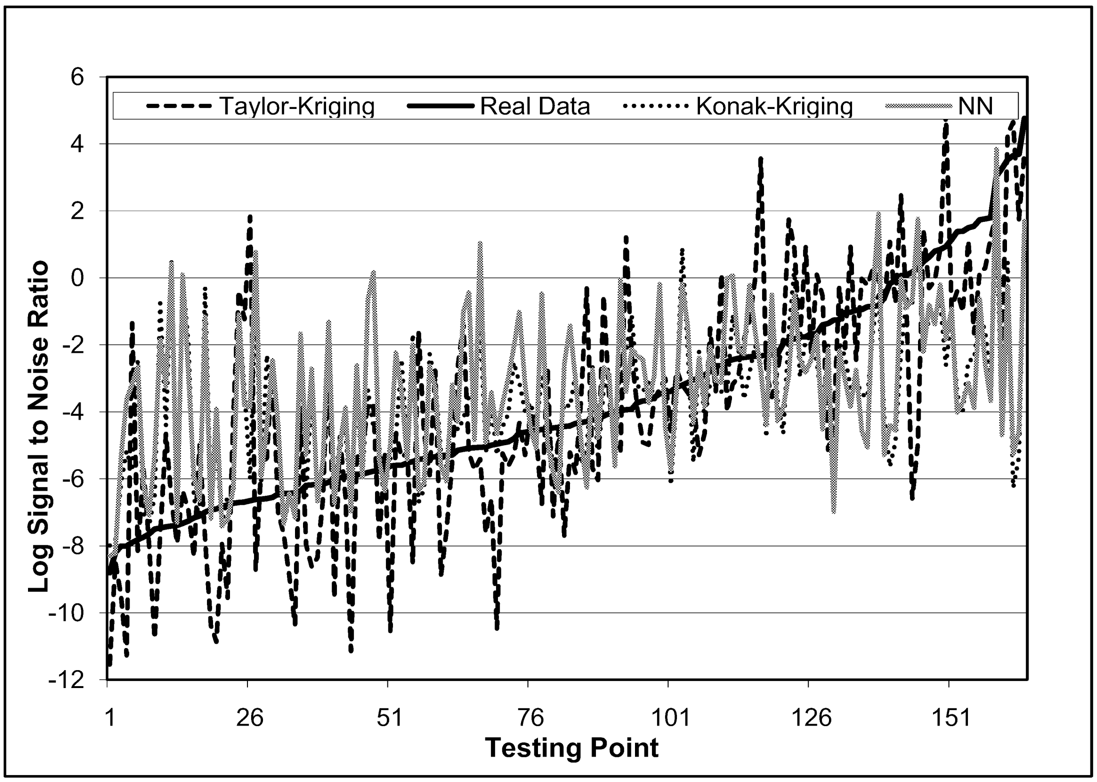


Figure 3: 27-Tower Wireless Network Comparison at 50% Training and 50% Testing

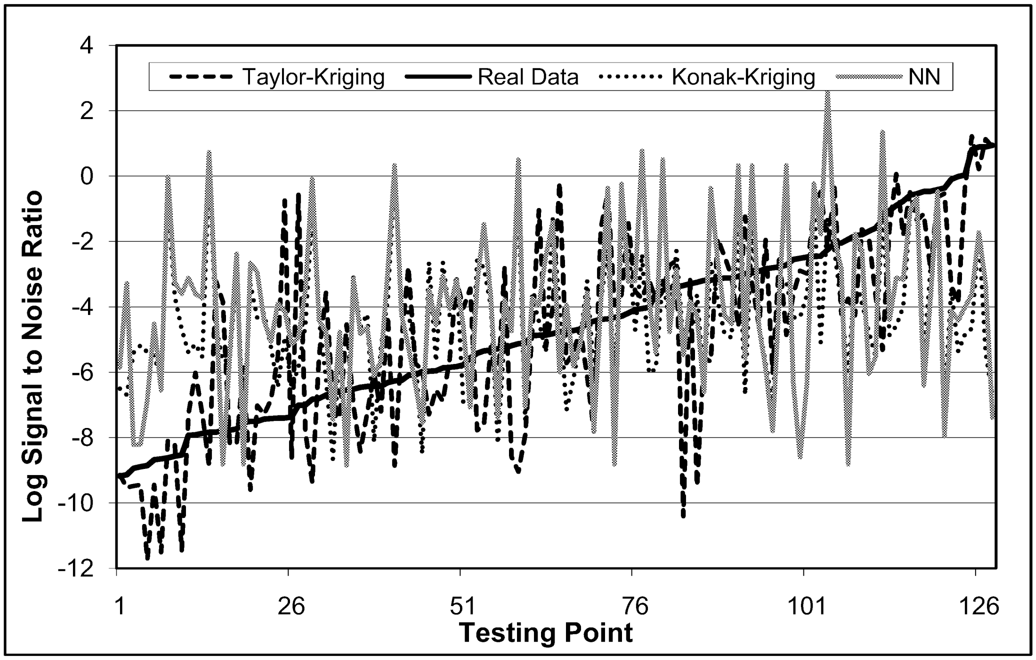


Figure 4: 45-Tower Wireless Network Comparison at 50% Training and 50% Testing

CONCLUSIONS

This simple study shows the promise of using Taylor Kriging to estimate signal strength for wireless networks using inputs of distance to tower and tower signal strength. Once calcluated for a given problem (set of towers and known signal strengths at some points) the Kriging model can be used to accurately estimate the signal strength for any point in the domain. The Taylor Kriging has the advantage of a single model and improved prediction ability for data sets with fewer training points over the application of Ordinary Kriging.

REFERENCES

Baioumy, S. M. H., Liu, H. P., Smith, A. E., 2008. "Application of Kriging to Cost Estimation." *9th Cairo University International Conference on Mechanical Design and Production*, Cairo, Egypt, 1546-1556.

Bartolacci, M. R., Konak, A. and Whitaker, R. M., 2004. "A Connectivity Decision Support System (CDSS) for Wireless Network," *Asian Journal of Information Technology*, Vol. 3, No. 9, 773-780.

Barton, R. R., 1994. "Metamodeling: A State of the Art Review." *Proceedings of the 1994 Winter Simulation Conference*. Ed. by Tew, J. D., Manivannan, S., Sadowski, D. A. and Seila, A. F., 237-244.

Chaveesuk, R. and Smith, A. E., 2005. "Dual Kriging: An Exploratory Use in Economic Metamodeling." *The Engineering Economist*. Vol. 50 (3), 247-271.

Forkuo, E. K., 2008. "Digital Terrain Modeling in Gas Enviroment". *International Archives of Photogrammetry, Remote Sensing and Spetial Information Science*, Vol. XXXVII. Part B2. Beijing.

Huang, D., Allen, T. T., Notz, W. I. and Miller, R. A., 2006. "Sequential Kriging Optimization Using Multiple-fidelity Evaluations." *Structural and Multidisciplinary Optimization*. Vol. 32 (5), 369-382.

Kleijnen, J. P. C. and van Beers, W. C. M., 2005. "Robustness of Kriging when Interpolating in Random Simulation with Heterogeneous Variances: Some Experiments." *European Journal of Operational Research*. Vol. 165 (3), 826-834.

Konak, A., 2009. "A Kriging Approach to Predicting Coverage in Wireless Networks". *International Journal of Mobile Network Design and Innovation*, in print.

Kumar, V., 2007. "Optimal Contour Mapping Of Groundwater Levels Using Universal Kriging - A Case Study". *Hydrological Science Journal*, Vol. 52 (5), 1038-1055.

Laslett, G. M., 1994. "Kriging and Splines: An Empirical Comparison of Their Predictive Performance in Some Applications". *Journal of the American Statistical Association*, Vol. 89, 391-400.

Liu H. P. and Smith A. E., 2007. "A Novel Particle Swarm Optimizer with Kriging Models." *2007 ANNIE Proceedings*, Vol. 17, 353-358.

Liu, H. P., 2009. *Taylor Kriging for Simulation Interpolation, Sensitivity Analysis, and Optimization*. Dissertation, Auburn University.

Leflbvre, J., Roussel, H., Walter, E., Lecointe, D. and Tabbara, W., 1996. "Prediction from Wrong Models: The Kriging Approach." *IEEE Antennas and Propagation Magazine*. Vol. 38 (4), 35-45.

Matheron, G., 1963. "Principles of Geostatistics". *Economic Geology*, Vol. 58, 1246-1266.

Nasereddin, M., Konak, A. and Bartolacci, M. R., 2005. "A Neural Network-Based Approach For Predicting Connectivity In Wireless Networks". *International Journal of Mobile Network Design and Innovation*, Vol. 1 (1), 18-23.

Oliver, S. and Gotway, C. A., 2005. *Statistical Methods for Spatial Data Analysis.* New York, CRC Press.

Yu, Y., Govindan, R., Rahimi, M. and Estrin, D., 2006. "Two Case Studies on Data Sensitivity of Wireless Sensor Network Algorithms and our Proposal on Scalable, Synthetic Data Generation." *International Journal of Distributed Sensor Networks*. Vol. 2, 355-386.

HEALTH
CARE
APPLICATIONS

VISUAL MODELING FOR MAKING HEALTHCARE SAFER

Ranjit Singh
Ashok Singh
Sonjoy Singh
Gurdev Singh

462 Grider Street, Clinical Center, Buffalo NY 14216. USA
e-mail: {gsingh4 / rs10}@buffalo.edu

ABSTRACT

Medical errors are a major cause of harm to patients. World Health Organization has formed an Alliance for Patient Safety. Improvements in safety can result through lessons learnt from error reports. These are a rich source for understanding of causes, cascades and consequences of errors. Calls for the development of appropriate error reporting and taxonomy systems that are useful at the point of care and policy levels are loud and clear. The urgency expressed in these calls presents a challenge and an opportunity to harness the power of computer visualization that can help structure and illustrate the "story" of an error in a universal language. This can overcome the shortcomings of current reporting methods and help create an unambiguous international error taxonomy. Presented here is a concept for web-based visual error reporting system. Although ambulatory care domain is used for illustration, this concept can provide a user-friendly, efficient means of reporting errors in any domain of healthcare. This unambiguous structured visual modeling is useful to all members of the health-care teams (including patients) at the point of care, as well as to policymakers.

INTRODUCTION

The huge chasm that exists between the potential and the actual quality of care delivered by the U.S. health care industry appears to be consistently wide across the nation [1]. It is reasonable to state that this chasm prevails across the world. According to The World Health Organization (WHO) patient safety is a Basic Human Right.

In the U.S., the Patient Safety and Quality Improvement Act of 2005 [2] is intended to encourage and facilitate error reporting. In conjunction with the President's 2004 call for national implementation of Electronic Medical Records (EMRs) and creation of the office of the National Coordinator for Health Information Technology, this should support the creation of searchable electronic databases of errors that are secure, involve low medico-legal risk, and can be analyzed and used to develop systemic solutions to healthcare safety problems [3].

Creation of a culture of safety is a critical first step for healthcare organizations that truly wish to improve quality and safety [4]. One of the steps in developing a culture of safety is the recognition by staff and clinicians of errors that occur on a regular basis [5]. One of the prime drivers to achieve this recognition is error reporting. Reporting systems need to be safe (that is, free from blame), easy, and worthwhile [6,7]. Error reports can be a rich source for understanding causes, cascades and consequences of errors, in turn leading to the design of interventions for improvement. It should be acknowledged that error reports are only the 'tip of the iceberg' since only a small fraction of errors are typically reported, and the information contained therein is limited to what reporters perceive and are willing to share. Other methods of analysis, including failure modes and effects analysis, root cause analysis, chart review, direct observation, and others, are needed to provide a more complete assessment of risks within an organization, Error reporting is nevertheless an important modality and should be seen as complementary to the other approaches.
Collation of reports into central databases can be useful at two levels. First, and currently the focus of most efforts, is the regional, national, or international level, which the authors shall refer to as the "macro-system level." These databases have the potential to receive large numbers of reports and therefore may be able to detect infrequent errors and track trends in reporting frequencies over time. In addition, since a large number of providers will, it is hoped, submit data, the publication of summary statistics will not compromise the confidentiality of individual providers. In the U.S., legislation will help to protect these data from medico-legal discovery [2].

Difficulty with this "macro-system level" error analysis is that the generalizations made about national data might not apply (or, be perceived by individual physicians to apply) to the individual practices or hospital floors. The Director of the U.S Agency for Healthcare Research and Quality (AHRQ) has emphasized that quality and safety information needs to be made useful at the point of care to patients and healthcare providers [8] . Similarly, the United Kingdom's House of Commons Committee of Public Accounts, in its report "A Safer Place for patients: Learning to Improve Patient Safety," calls for a unified and convenient form for reporting and taxonomy that encourages feedback on solutions to *specific* patient safety incidents [9]. Therefore, in addition to the "macro-system level" data, individual practices/healthcare-sites and organizations need local "micro-system level" information that is directly relevant to them and can be used internally to drive safety improvement. Such information, reported internally for quality and safety improvement purposes, potentially has more legitimacy in the eyes of local staff and clinicians in any health care setting.

The general purpose of this work is to develop and test a concept for a visual medical error taxonomy, built on visual reporting, that can provide for both "macro-system" and "micro-system" level needs. Figure 1 depicts the overall concept in which error reporting at the micro-system level is used internally for safety improvement as well as being fed seamlessly to a regional, national, or international database that is used to study the epidemiology of errors and to generate alerts. The purpose of this paper is to present the concept of visual reporting. Before presenting this concept it will be helpful to describe the framework of the error taxonomies that have to be populated by the proposed visual reports.

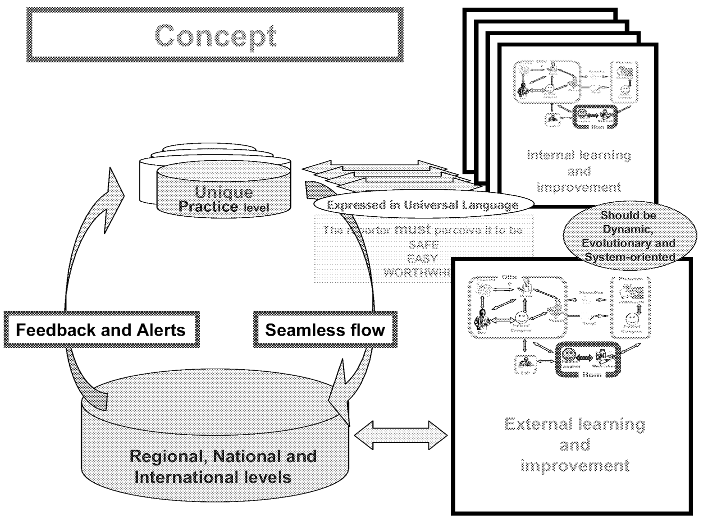


Figure 1. Overview of the web-based concept for visual taxonomy and reporting.

ERROR TAXONOMIES

Numerous error taxonomies have been and are being developed to organize and classify error reports. The IOM's report "Patient Safety: Achieving a New Standard for Care" [10] calls for the development of an event taxonomy. The World Health Organization (WHO) [11] is working to establish a common international system for classification. The International Primary Care Patient Safety Taxonomy Steering Committee has set itself the important and necessary task of developing "*a primary care taxonomy for patient safety, embedded in the International Classification of Primary Care (ICPC-2) and in an episode of care structure, that can operate across settings and vendors, and that maps to other standards and data structures.*" [12].

Current taxonomies are essentially alpha-numeric codes that are used to classify error data and summarize it (whether at local, regional, national and international levels) for various purposes including:

- Communication of information about errors and their characteristics including causative factors, consequences, and severity (keeping in mind that error reporting alone may be insufficient for fully addressing these issues)
- Estimation of frequencies and trends of various error types
- Identification of needs for safety improvement

These taxonomies have some limitations:

1. The coding systems are complex and prone to ambiguity
2. They do not readily meet the *point-of-care* needs of patients and health providers to understand, within their own unique micro-systems, the causes, cascades and consequences of the reported errors
3. They do not fully capture the "story." By reducing an incident to a series of codes, the flavor of the event is lost. It is the "story" that has the greatest potential to contribute to safety improvements [6,13]
4. They often differ in the way they define, count and track events, and they use different terms, data and coding methods and analysis. This makes it difficult to compare data that have been collected or coded using different taxonomies

The U.S. IOM [10] states that a comprehensive National Health Information Infrastructure must provide information flow across three dimensions: (1) personal health, to support individuals in their own wellness and health decision making; (2) health care providers, to ensure access to clinical decision support systems; and (3) public health, to address and track public health concerns and health education campaigns. Items (1) and (2) correspond to the micro-system level while item (3) is at the macro-system level. Use of a consistent error taxonomy across these levels is imperative.

The imperatives for consistent error taxonomy at both micro- and macro-system levels presents an opportunity to harness the benefits of computer visualization. The authors' experience with visualization so far suggests that this helps to create crosswalks across disparate taxonomies. A very important feature of visualization is that it can help to structure and illustrate the "story" of an error or event. The proposed visual taxonomy is coded at four main levels, corresponding to the structure of the visual models [14]:

- Health care domain
- Process
- Sub-process
- Entity/interaction

An event can consist of one or more errors, together with causes and consequences. Each of these is coded at the above four levels.

VISUALIZATION

The authors take the view that visualization is a universal tool that furnishes a natural common "language." For instance, it is used effectively for international road signs. It respects and aids inductive (as against linear) perception and decision making. It can provide:

1. a fast path to fully engaging the minds of individuals and their teams including patients
2. insight to causes, cascades and consequences of errors
3. a common vision for teamwork, with the potential for improved outcomes

4. aid for coping with the complexities, fragmentation and decentralization of the health care system[2] and
5. aid for mapping across different taxonomies and data structures [15].

Applying a systems-engineering/management approach, the authors have developed visual models at the macro-system and micro-system levels [16].

MACRO-SYSTEM MODEL

The macro-system model is a high-level view (Figure 2) of the health-care system. The processes of care are represented by the radials. These processes are recognized to occur in a cyclical fashion as shown by the clockwise progression around the circle from *Assessment* to *Plan* to *Implementation*, *Feedback*, *Review & Learn* and back to Assessment again [14, 16].

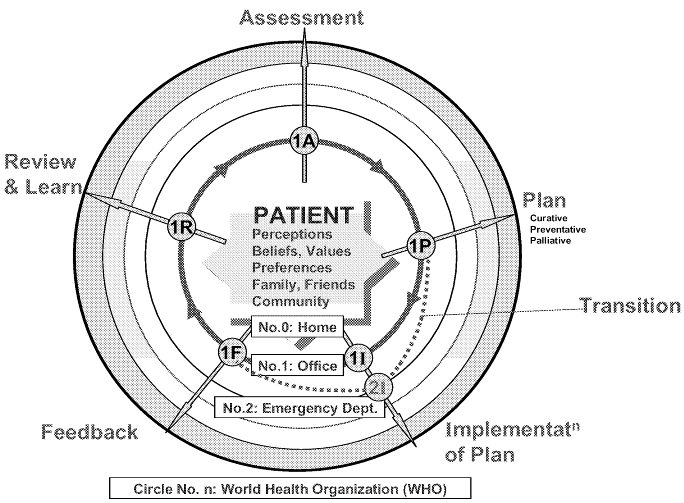


Figure 2: Macro-system model of health care.

These cycles of care take place in various domains that are depicted by concentric circles. The increasing sizes of the circles depict the enlarging involvement of the system, starting from the patient level (Circle No. 0) at the center to the international health authority level (Circle No. n) on the outside. The innermost circle represents the patient in his/her own domain (i.e. Home/Community) and recognizes that this is the place where most 'health-care' actually occurs. International health authorities (e.g., World Health Organization), depicted by the outermost circles, play an important role in devising public health policies that can impact management of patients at all points within the system. Office-based primary care is represented by circle No.1. Depending on the system under study, circle No. 2 might represent the emergency room and No. 3 might represent the hospital inpatient setting, etc.

The main purpose of this macro-system model is to understand a patient's care in the context of the overall healthcare system, especially with respect to errors and opportunities for errors, including in transitions between different parts of the system.

Cycles of care can occur multiple times in one setting and/or involve transitions between settings. The macro-level view aims to provide the 'big picture' so as to facilitate understanding of the processes of care in different interrelated parts of the system and transitions between these parts, helping the user to understand interdependencies and needs for information flow.

MICRO-SYSTEM MODELS

Micro-system models are close-up views of the system; each may represent one or more points within the macro-system model. For example, one may devise a micro-system model for a specific domain within the macro-system, or for a specific process within a domain. These models show how the various entities/agents in the micro-system interact. The level of detail represented in a micro-model depends on the purpose for which it is used. Figure 3 is an example of a micro-system model for medication management in ambulatory settings and shows activities in the office, pharmacy, home, laboratory, imaging/radiology facility, and third party payer, and the interactions within and between these. Each interaction is shown as an arrow. Errors or safety problems can originate at any, or at multiple points within the system.

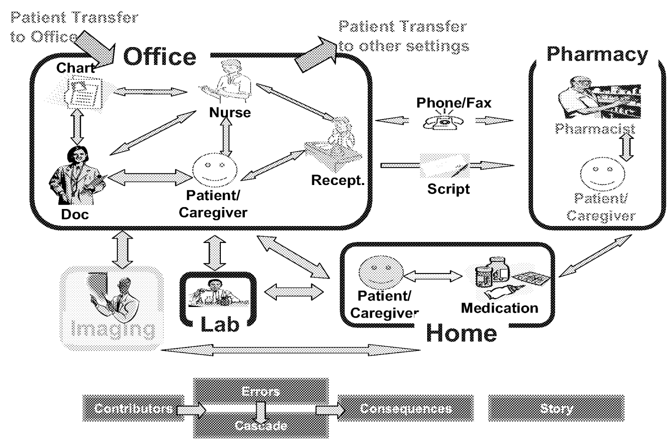


Figure 3: Micro-system model of a primary care setting.

The macro-system and micro-system diagrams are computerized and contain 'hyperlinks' that facilitate hierarchical linkage between models and can be used for dynamic data links within databases. For example, any point on the micro-system model can be linked electronically to a table containing relevant data about errors that are known to occur at that point in the system with details of frequency and consequences of these errors as well as corrective action recommended or used. These macro- and micro-system models can also provide various other functions that the authors have described elsewhere [16].

A VISUAL ERROR REPORTING TOOL

Figure 4 is an example of how a visual reporting tool could be used, based on the same micro-system model shown in Figure 3. To report an error, the user would first describe the patient's demographic details and enter other information deemed appropriate, such as their job designation, circumstances in which they discovered the error, etc. Then they would commence entering details of the error using the visual interface.

In this case, the error is that the primary doctor (who is reporting this error) refilled the wrong dose of a blood pressure medication by phone. The patient is a 76-year-old female with type 2 diabetes mellitus, hypertension, and coronary artery disease (CAD). She sees her primary doctor every 3 months and is on various appropriate medications, including quinapril 10 mg daily for hypertension. She also sees a cardiologist annually for CAD followup and management. At today's visit to the primary doctor's office, the doctor notices that her blood pressure is above goal at 147/90, while it had been well controlled at previous visits (including the most recent visit 3 months ago). Therefore, he/she inquires as to the patient's compliance with the medication, to which the patient replies "my pressure's probably up because you cut down my medication dose last time." The doctor reviews the chart and finds no documented change in any blood pressure medication. He/she inquires further and discovers that at the patient's previous visit to the cardiologist (8 months earlier), the cardiologist had noted elevated blood pressure and increased the dose of quinapril from 10 mg to 20 mg daily and also prescribed a 6-month supply. Then, 2 months ago, when the patient was running out of quinapril, she called her primary doctor's office for a refill. The doctor reviewed the chart and instructed the nurse to phone in a prescription for quinapril 10 mg daily, since this was the dose documented in the patient's chart. There was no consult letter in the chart from her cardiologist. The patient had seen the primary doctor twice since the cardiology visit but apparently had not mentioned the dose change.

Panel 1 of Figure 4 shows how the doctor would indicate the location of the error, which in this case is in the communication (via telephone) between the doctor's office and the pharmacy. Next, in Panel 2, when presented with a list of possible errors in this step, the reporter picks the relevant item from the list, which in this case was "Wrong dose." Next, the user chooses to describe the contributing factors. As mentioned earlier, one of these was that the chart did not contain any information from the cardiologist regarding the dose change. The user therefore clicks on the chart and chooses the appropriate item from the list, as shown in Panels 3 and 4. Another contributor was that the patient did not inform the primary doctor about the dosage adjustment; this can be entered in the same fashion.

Similarly, the user is prompted to indicate the location and nature of any consequences. In this case (Panels 5 and 6), the patient was under-medicated. Finally, the severity of the error can be elicited, usually on a scale, as indicated in Panel 7, and the user types a brief narrative description of the event to add any other details and help to eliminate any ambiguities (Panel 8).

The various lists, hyperlinked to the entities and their interactions, are designed to help reduce emotive and cognitive biases in perceptions and reporting.

DISCUSSION

The authors have proposed a novel approach, based on computerized visual models of the health care system, to facilitate the reporting, summarizing, and dissemination of information about medical errors in healthcare. The purpose is to make information about medical errors useful both at the practice level and at the policymaking level. The ability to view a macro- or micro-system diagram together with error frequency information can be valuable in helping decision makers at various levels in the health care system identify and prioritize areas for system improvement. Similarly, the ability to summarize a single event—including errors, contributing factors, and consequences—in a clear visual format would appear to provide some advantages when compared to a list of codes. It should be noted that in any reporting system, reports are submitted by human beings who have their own unique viewpoints and past experiences that color their perception of incidents. For example, perceptions of contributing factors will likely vary among reporters for the same incident.

The authors take the view that visual modeling can help overcome this issue because the process of reporting involves looking at and interacting with system models. These remind the reporter of the processes that are in place, his/her role in them, the problems that can occur, contributors that might be present, and consequences that can occur, thereby improving situational awareness [18], as well as aiding narration of the "story." In other words, the visual models and associated drop-down lists have the potential to help create a common vision of the system. Furthermore, the authors suggest that a visual format can facilitate information sharing with team members and other stakeholders (including patients and families) and has the potential to enhance the understanding of events, thus facilitating the development of appropriate preventive strategies.

Error reporting using a visual format would require appropriate staff training. Staff would need to be familiarized with the visual models and the interface. However, some of the potential benefits outlined above justify such an up-front effort.

Another benefit, important from a practical perspective, is the fact that this visual reporting approach allows the user to code the error while reporting it. This contrasts with conventional reporting systems using existing taxonomies, which require considerable time and effort to dissect written error reports and code them. Individual healthcare settings wishing to collect and understand local error data generally cannot afford the time and effort required to manually code errors using alpha-numeric taxonomies, nor are they likely to have the expertise to do so.

Further work is needed to fully 'operationalize' the concepts described here and to evaluate the usability of the visual interface and its potential benefits. In order for the process to be used across all health care settings and internationally, it would be necessary to create visual diagrams of other systems. The Authors are beginning to create standardized icons for the whole range of entities in the various settings of the health care system. These would enable interactive creation of micro-system models (potentially by end-users) for any setting.
Figure 5 shows two examples of micro-system models developed for falls and postoperative pain management in

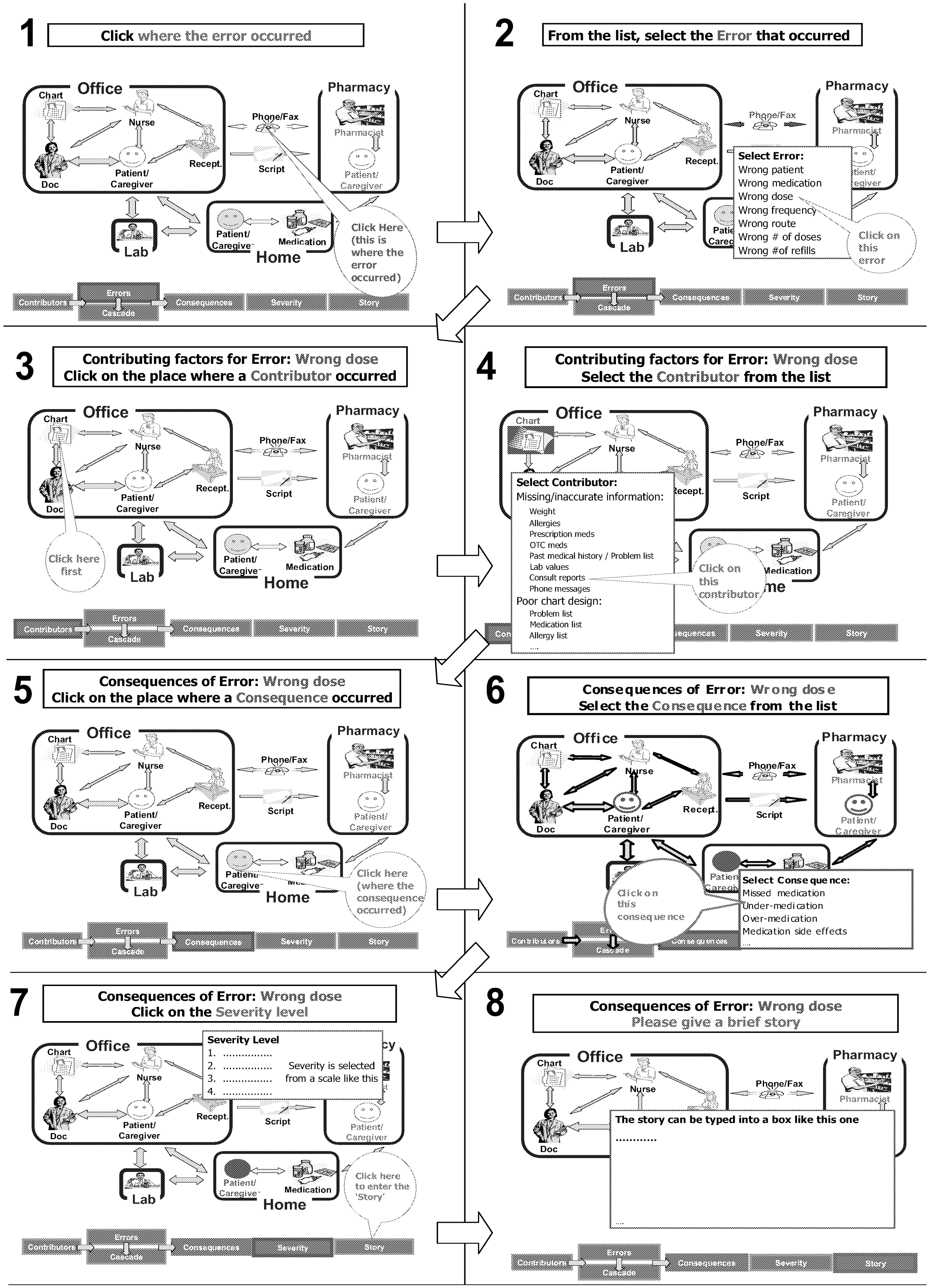


Figure4: Example of Interactive Error Reporting in an Ambulatory Setti

hospital setting. In addition, to facilitating use in a wide variety of settings, this kind of reporting tool should be accessible directly from within electronic medical record systems and should be able to import patient data directly from these records. A recent study [19] in the domain of operating rooms demonstrated that integration of an incident reporting system into an electronic patient record significantly increased the number of incidents reported.

While tracking rates of errors over time or comparing rates among different institutions or regions are commonly perceived aims of error reporting systems, caution is needed in interpreting such data because of the problem of underreporting. According to IOM estimates, only about 5 percent of known errors are reported [2]. Therefore, differences in rates of errors reported over time or among institutions do not necessarily reflect true differences in rates of errors but may merely represent differences in reporting behavior. In the USA Patient Safety Organizations (PSO) are being setup to receive confidential and anonymous error reports. Figure 6 portrays the model proposed by the authors to facilitate the whole process at not only healthcare-domain, regional and national levels but also at the WHO level.

It is important to point out that those errors that are reported most frequently are not necessarily the errors that occur most frequently. They are merely the ones that reporters feel more comfortable reporting [20]. It is hoped that creating more user-friendly and intuitive reporting tools, such as the one described here, will help increase reporting rates and so, provide more opportunities to learn. However, this needs to be done in concert with changes in organizational culture [21] that encourage reporting and learning from errors and discourage blame and punishment for errors that are due to systemic problems. In other words, a shift from the prevailing culture of blame to a culture of safety is called for.

References

1. McGlynn EA, Asch SM, Adams J, et al. The quality of health care delivered to adults in the United States. N Eng J Med 2003; 348: 2635-2645.
2. Patient Safety and Quality Act, United States Public Law 109-41, 2005. Available at: www.mihealthandsafety.org/pdfs/pub-3-scheffler.pdf. Accessed May 7, 2009.
3. Thompson TG, Brailer DJ. The decade of health information technology: Delivering consumer-centric and information-rich health care: Framework for strategic action. Washington, DC: Department of Health and Human Services; 2004.
4. Kohn LT, Corrigan J, Donaldson MS, eds. To err is human: Building a safer health system. Washington, DC: National Academies Press; 2000.
5. Joseph GM, Stee L, Marian TM, et al. Journal reporting of medical errors: The wisdom of Solomon, the bravery of Achilles, and the foolishness of Pan. Chest 2007; 131: 890-896.
6. Billings CE. Incident reporting systems in medicine and experience with the Aviation Safety Reporting System. A tale of two stories: Contrasting views of patient safety. Chicago: National Patient Safety Foundation at the AMA; 1998.
7. Leape LL, Arbookire S. WHO draft guidelines for adverse event reporting and learning systems. Geneva, Switzerland: World Alliance for Patient Safety; 2005.
8. Clancy CM. Evaluating the potential of new technology. Building a better delivery system: A new engineering/health care partnership. Washington, DC: National Academies Press; 2005.
9. House of Commons Committee of Public Accounts, A safer place for patients: Learning to improve patient safety, HC 831. London: The Stationery Office Limited; 2006.
10. Aspden P, Corrigan JM, Wolcott J, et al, eds. Patient safety: Achieving a new standard for care. Washington, DC: Institute of Medicine, Committee on Data Standards for Patient Safety; 2004
11. Chang A, Schyve PM, Croteau RJ, et al. The JCAHO patient safety event taxonomy: A standardized terminology and classification schema for near misses and adverse events. Int J Qual Health Care 2005; 17: 95-105.
12. International classification for patient safety. Geneva, Switzerland: World Health Organization. Available at: www.who.int/patientsafety/taxonomy. Accessed May 7, 2009.
13. Chen F, Philips R. NAPCRG leads efforts to distribute ICPC information. North American primary care research group newsletter. 2005 February; 33.
14. Weingart SL. Beyond Babel: Prospects for a universal patient safety taxonomy. Int J Qual Health Care 2005; 17: 93-94.
15. Singh R, Pace W, Singh S, et al. A concept for a visual computer interface to make error taxonomies useful at the point of primary care. Informat Prim Care 2007; 15: 221-229.
16. Buzan T. The mind map book. New York: Penguin; 1991.
17. Singh R, Singh A, Fox C, et al. Computer visualization of patient safety in primary care: A systems approach adapted from management science and engineering. Informat Prim Care. 2005; 13: 135-144.
18. Craig PA. Controlling pilot error: Situational awareness. New York: McGraw-Hill; 2001.
19. Haller G, Myles PS, Stoelwinder J, et al. Integrating incident reporting into an electronic patient record system. J Am Med Inform Assoc 2007; 14: 175-181.
20. Nuckols TK, Bell DS, Liu H, et al. Rates and types of events reported to establish incident reporting systems in two U.S. hospitals. Qual Saf Health Care 2007; 16: 164-168.
21. Singh R, Singh A, Taylor JS, et al. Building learning practices with self-empowered teams for improving patient safety. J Healthc Manag 2006; 8: 91-11

BIOGRAPHIES

Ranjit Singh MB BChir, MA (Cantab), MBA, MABFM, MD, a graduate of the University of Cambridge (England) where he read Medicine and Management. He is Associate Director and Professor, Patient Safety Research Center, State University of New York at Buffalo, USA. He has authored numerous papers

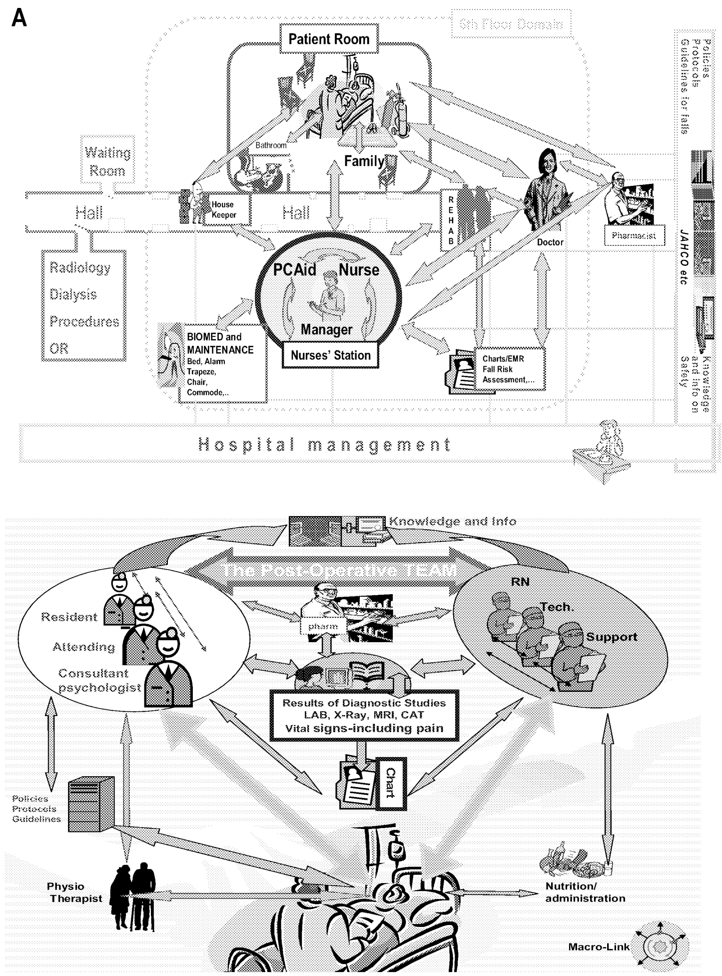


Figure 5: Examples of Micro-system Models for Falls (A above) and Postoperative Pain Management

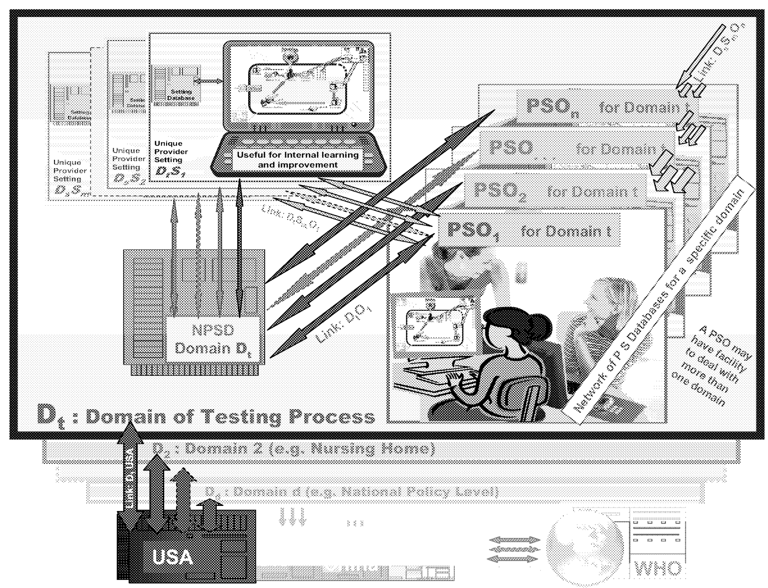


Figure 6: Potential Links Between PSOs, National and International Data Bases

on patient safety and simulation and served on the US National Expert panel on reporting in primary care.

Ashok Singh, MB BChir, MA (Cantab), MABFM, MD, is a graduate of the University of Cambridge (England) where he read Medicine and Management. He is a Partner of The Niagara Family Medicine Associates, Niagara Falls. He has authored papers on patient safety and simulation.

Sonjoy Singh MB BChir, MA (Cantab), MRCGP, MABFM, is a graduate of the University of Cambridge (England) where he read Medicine. He is a Partner of The Niagara Family Medicine Associates, Niagara Falls, NY, USA. He is recipient of Western New York "Top Doctors" awards and has authored papers on patient safety and simulation.

Gurdev Singh MScEng PhD (Birm), is the founding Director of Patient Safety Research Center at State University of New York at Buffalo, USA. In 1996 he was recognized by the US Department of Justice as an 'Alien of Extraordinary Ability.' This classification is awarded to a person who has demonstrated that s/he has reached the top of her/his profession at the international level for a sustained period of time, or to a Nobel Laureate or equivalent. His experience and themes of his publications include: development and use of hybrid decision support systems, multidisciplinary appraisal of risk and reliability in projects and structures based on Monte Carlo simulation, multi-objective and multi-resource optimization and 'satisfisation', development of Singh-Markowitz efficient frontier method, concurrent engineering, stochastic and visual simulation of construction projects, and computer-aided learning of risk and reliability analysis.

WEB-BASED SYSTEMS APPROACH TO IMPROVEMENT OF PATIENT SAFETY

Ranjit Singh
Ashok Singh
Chimay Anumba
Gurdev Singh
462 Grider Street, Clinical Center, Buffalo NY 14216. USA
e-mail: {gsingh4 / rs10}@buffalo.edu

ABSTRACT

According to The World Health Organization patient safety is a Basic Human Right. It has, therefore, formed an Alliance for Patient Safety. The reason is that medical errors are a major cause of harm to patients. The authors support the assertion by the US Institute of Medicine that there can be no quality of care without patient safety. The unique nature of each healthcare setting means that 'off-the-shelf' top-down solutions seldom work. The purpose of the work reported here therefore was to develop a bottom-up methodology in which staff is empowered through IT-assisted systems approach. This team-based approach is based upon an Error Reduction Intervention Cycle as follows: (i) an anonymous on-line survey in which all staff in a healthcare setting rate various errors according to their perceived frequency and severity; (ii) identification of priority areas based on hazard ratings derived from the survey results (supplemented by Delphi technique); (iii) development of solutions to the prioritized problems; (iv) implementation of the solutions, and tracking their effect/s, and (v) repetition of the survey for continuing safety-based quality improvement. The tool is highly adaptable for any health care setting. For the purposes of illustration the case of primary care settings, where the vast majority of any nation's healthcare takes place, is presented.

INTRODUCTION

Safety and quality of care are high on the agendas of government, payers, providers, and increasingly of patients and their families [1,2]. However, relatively little is known about the factors that pose the greatest threat to safety-based quality of care in various settings such as different specialties in hospitals and primary care offices. The huge chasm that exists between the potential and the actual quality of care delivered by the U.S. health care industry appears to be consistently wide across the nation [3]. It is reasonable to state that this chasm prevails across the world.

The paradigm of complex adaptive systems suggests that each medical setting can be viewed as a complex adaptive micro-system; to thrive, such a micro-system needs to identify its own unique set of problems and devise solutions that are tailored to the situation, in light of the current quality status, practice costs, and resources available [4-6]. To achieve this, the measurements that are used to identify and prioritize quality and safety problems must be trusted by the members of the system. The literature reflects skepticism regarding externally driven measures[7-9] and top-down recommendations for improvements, suggesting that, in their current forms, they may not be trusted by many physicians, nurses and other staff as fair and valid measures. One approach to identifying problems internally has been to use error reports. The intention is that all the staff members complete these documents voluntarily when they notice mistakes or adverse events and that the information is used internally to drive improvement. Error reporting systems are widely available in hospital and ambulatory settings (commonly under the title " incident reports "), but their usefulness has always been limited by the problem of underreporting [1]. An alternative approach that permits involvement of all team members to identify and prioritize safety and quality problems is Failure Modes and Effects Analysis (FMEA) [10]. This has been widely used in other high-risk industries and has been advocated by the Institute of Medicine [1] as a means of analyzing a system to identify its weaknesses (failure modes) and possible consequences of failure (effects) and to prioritize areas for improvement. The Joint Commission (that accredits Healthcare Organizations in the USA and elsewhere) has required since 2002 that all accredited hospitals perform proactive risk assessment each year following a series of steps based on FMEA [11,12]. It is important to point out that this method is based on the perceptions of informed decision makers. This process, usually made up of 8 steps [12], is time consuming, costly, and requires considerable expertise and experience. In hospital settings, trained quality improvement personnel are sometimes available, and leadership is mandated by the Joint Commission to provide the necessary resources for this type of activity. In most and particularly ambulatory settings, however, these factors typically are not present, and the literature is sparse regarding use of FMEA in all settings. In rural ambulatory settings, the scarcity of the necessary resources and expertise is particularly problematic in the US and more so in the industrially challenged countries [3].

In an attempt to overcome some of these practical barriers while maintaining the essential thrust of FMEA for all healthcare settings, the authors have developed an adapted FMEA process. This paper describes that process and their experiences in various settings. The adapted FMEA process involves administration of an anonymous safety survey to practice members followed by analysis of the results in order to develop a priority list of safety threats.

SYSTEMATIC APPRAISAL OF RISK AND ITS MANAGEMENT FOR ERROR REDUCTION (SARAIMER) IN HEALTHCARE

The *science of observed systems* is looped with that *of observing systems* in the development of SARAIMER.

The resulting cybernetic loop forms the integral part of a team-based systems approach developed by the authors, and it is termed "Error Reduction Intervention Cycle" (ERIC) comprising of four stages:

(i) an anonymous on-line survey in which all staff in a healthcare setting rate various errors according to their perceived frequency and severity;
(ii) identification of priority areas based on hazard ratings derived from the survey results (supplemented by Delphi technique);
(iii) development of solutions to the prioritized problems;
(iv) implementation of the solutions, and tracking their effect/s, and
(v) repeat survey for continuing safety-based quality improvement.

The overall methodology is illustrated in Figure 1.

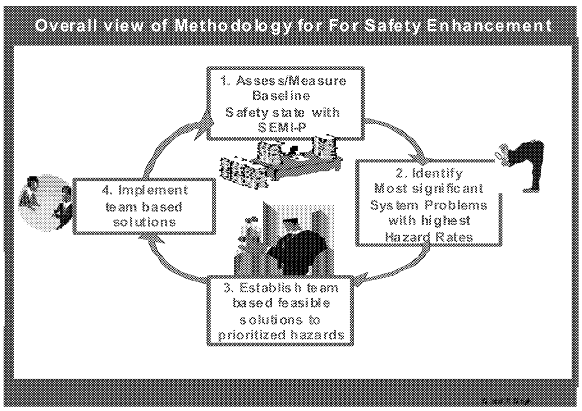


Figure 1: The Cyclic Safety Improvement Methodology

Stage (1), Anonymous on-line Survey: This is performed with an instrument termed "Safety Enhancement and Monitoring Instrument that is Patient centered" (SEMI-P). This instrument is designed to minimize the cognitive and emotive biases in respondents. The first step in this adapted FMEA process is to begin to understand the system of care in the setting. This was done by first identifying the various entities in the practice (such as the patient, provider, nurse, and chart), listing the main interactions between them, and then portraying them in a diagram (Figure 2)[13].

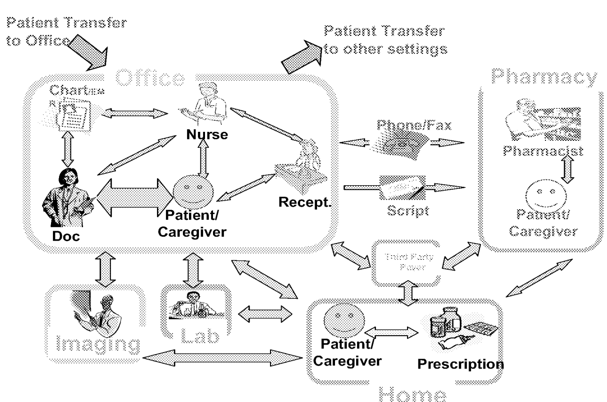


Figure 2: Micro-system model of a primary care setting.

This diagram is then used as the basis for an anonymous SEMI-P survey. Errors can occur at any point in the setting, including within entities and in the interactions between entities. The survey is designed to examine 12 key areas, with one page of the survey dedicated to each (eg, the nurse-provider interaction). Each page consists of a list of failure modes (errors or causes of error) that can occur in that part of the system. The lists, which include a total of 140 different failure modes, were developed after a review of the literature [1,14-20] and consultation with the practicing physician and nursing leaders. The lists can be customized to incorporate special circumstances for any given setting if desired. Figure 3 shows part of an example page from this instrument. Participants are asked to consider each of the listed errors in turn and, for each, to respond with their perception of the frequency of the error and the likely severity of the consequences. Explanations of the categorical choices are given at the bottom of each survey page. The diagram of the practice entities is included on each survey page, with the appropriate entities highlighted to orient respondents to the part of the setting being assessed. All members are asked to complete the survey anonymously, stating (if willing) the group to which they belonged (provider, nursing, or administrative).

Stage (2), Hazard Analysis for Identification of most hazardous failure modes and prioritization: The goal of the analysis was to rank the failure modes from the survey according to the size of their effects, as perceived by practice staff. We chose to measure effects using the concept of hazard borrowed from engineering. The hazard posed by any given error (or failure mode) is equal to the frequency (ie, probability) with which it occurs multiplied by the severity of the consequences that accrue when it does occur (hazard = frequency × severity). In the survey, each respondent rates the frequency and severity of each error according to the categorical scales shown in Figure 3.

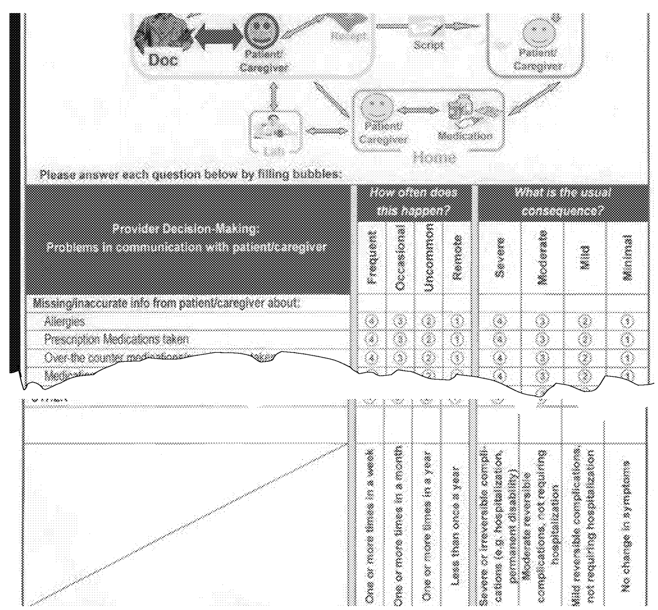


Figure 3: An example page from a multi-page SEMI-P.

The categorical responses from stage (1) are converted to numerical values so that hazard values could be calculated, summarized, and compared. The first step in this process is to calculate frequency in terms of each setting's patient volume, yielding a rate per 1,000 patient visits. For example, the "frequent " probability of occurrence (which was described as 1 or more times in a week) was approximated as 2 times per week. For example if 220 patients were seen per week the " frequent " was converted to 2/220 = 0.009, or 9 times per 1,000 visits.

Similarly the " remote " probability, described as less than once a year, was approximated to 0.5 times per year and calculated as 0.5/11,000 (patients seen per year) = 0.00005, or .05 times per 1,000 visits. For the severity scores, conversion of the categorical responses to numerical equivalents required decisions regarding the relative weights of the outcomes. For example, how much worse is a "moderate" outcome than a "mild" outcome? In utility theory [21], the severity outcome can be seen as a loss of utility, or disutility. In health care settings, risk aversion is most appropriate as it gives proportionately more weight to the most severe outcomes, that is, those that are most important to avoid. Based on this, the following numerical values were assigned: minimal = 0.01, mild = 0.05, moderate = 0.2, severe = 1.0. Sensitivity analysis using different values, maintaining the risk aversive assumption, showed that hazard rankings remained stable, suggesting that the precise values chosen are not critical to the process.

Hazard scores are then calculated for each failure mode for each respondent by multiplying the numerical conversions for frequency and severity. Mean hazard scores are also calculated for each respondent subgroup (providers, nursing, administrative) and for each practice site as a whole. To allow analysis of concordance between practices, hazard values are scaled to give a range of 0.01-100. The hazard scores are also listed in rank order (highest to lowest) for each subgroup and practice. Table 1 shows the hazard matrix used. As stated earlier, this is based on the fact that the healthcare workers and patient are risk aversive.

Table 1: Hazard Matrix

Calculation of Hazard: *h = p x s*

Hazard Matrix

'button' Nos,

	Probability (p)			
Severity (s)	1 Remote	2 Uncommon	3 Occasion	4 Frequent
Minimal 1	0.01	0.02	0.24	1
Mild 2	0.03	0.10	1.20	5
Moderate 3	0.10	0.40	4.80	20
Severe 4	0.50	2	24	100

The software was written to provide hazard results in a graphical format to facilitate understanding of the system vulnerabilities. This was followed by web-based consensus forming process (Delhi Technique) the results of which were also presented to staff in a graphic format. This resulted in team decision to select the hazards that needed to be addressed.

Stage (3), Development of Solutions: In light of the resources available and the capabilities of the unique setting the teams develop solutions to address the prioritized vulnerabilities. These solution are informed by the established safety principles and strategies.

Stage (4), Implementation of the improvement interventions and tracking: The staff is helped to form implementation teams with clear allocation of responsibilities and time schedule. Along with this, the outcome of the intervention is tracked with the aid of the software written for the purpose.

DISCUSSION

Each healthcare setting is a unique and complex micro-system and should be respected and treated as such. This means that off-the-shelf solutions seldom work. These settings must be helped to focus their limited resources on solving safety problems that are most hazardous rather than following the common and convenient practice of simply addressing those that are most amenable to solution.

Presented in this paper is a novel approach, adapted from the method of FMEA intended to be used by each unique healthcare setting to identify its own set of priorities and to internally develop feasible solutions based on the team members' intimate knowledge of their micro-system. Figure 4 illustrates the systems methodology. Field applications have demonstrated that the diverse members of the setting teams were able to work together to develop and implement solutions. A further intended effect of the methodology presented here is the fostering of a culture of safety. *The generic nature of the proposed methodology lends itself to its use for quality improvement of any service or production organization.*

The ERIC process aims to incorporate change management strategies through a motivated guiding coalition of all staff with a clear, shared vision and shared goals [23]. Empowerment, ownership, good team formation (that fosters mutual respect, trust, understanding, collaboration, cooperation, and work satisfaction) are sought to be the driving "strange attractors" of a learning, self-directed, adaptive, and evolving organization [23, 24, 25] leading to a culture of safety. This philosophy contrasts with the prevalent approach, often referred to as "Taylorism," (described by Frederick Taylor in his influential 1911 book, *The Principles of Scientific Management*) [26] in which, in its extreme form, only management is empowered to make decisions while workers are expected to follow unquestioningly.

The field tests with SARIMER in the domains of postoperative-pain-management, inpatient-falls-management and primary care settings suggested that the approach was well received by all the staff. High response rates were achieved with minimal disruption of activities. The analysis has been automated using specially developed web-based facility for prioritization through Delphi technique. The fact that team members worked constructively together, with or without computerization of SEMI-P, is one sign of staff empowerment and a development of safety culture.

Within each setting, there was a moderate level of agreement between subgroups of respondents regarding the top 10 sources of hazard. Convergence of opinions within a setting was helpful since it showed that team members were "on the same page". On the other hand, variations in opinion are to be expected since different subgroups and different individuals within those subgroups will each have their own perspectives. These similarities and differences of opinion should be embraced and explored by setting members to help in mutual understanding. The level of concordance between primary care settings was lower than that seen within these settings, lending support to our assertion that each practice is unique. The proposed approach is based on the anonymously expressed perceptions of individual members of the practice team and can be seen as an attempt to capture the memory of the whole practice and foster teamwork.

CONCLUSION

The Web-based SARAIMER methodology shows promise as an approach to identifying the most serious threats to patient safety in any healthcare setting by following the **e**rror **r**eduction **i**ntervention **c**ycle (ERIC) using the **s**afety **e**nhancement and **m**onitoring **i**nstrument that is **p**atient centered (SEMI-P).

REFERENCES

1. Kohn LT , Corrigan J , Donaldson MS , eds . *To Err Is Human:Building a Safer Health System* . Washington, DC : National Academy Press ; 2000.
2. Institute of Medicine . *Crossing the Quality Chasm: A New Health System for the 21st Century* . Washington, DC : National Academy of Sciences ; 2001 .
3. Institute of Medicine Board on Health Care Services . *Quality Through Collaboration: The Future of Rural Health Care* . Washington, DC : National Academies Press ; 2005.
4. McDaniel RR Jr . Strategic leadership: a view from quantum and chaos theories . *Health Care Manage Rev.* 1997; 22 : 21 - 37.
5. Miller WL , Crabtree BF , McDaniel R , Stange KC. Understanding change in primary care practice using complexity theory . *J Fam Pract* . 1998 ; 46 : 369 - 376.
6. Plsek P . Redesigning health care with insights from the science of complex adaptive systems . In : *Committee on Quality of Health Care in America, Institute of Medicine, ed. Crossing the Quality Chasm: A New Health System for the 21st Century* . Washington, DC : NationalAcademy Press ; 2001 ; Appendix B : 309 - 322 .
7. Rosenstein AH . Hospital report cards: friend or foe? *J Clin Outcomes Manage* . 2004 ; 11 : 98 - 102 .
8. Hibbard JH , Stockard J , Tusler M . Does publicizing hospital performance stimulate quality improvement efforts? *Health Aff* . 2003 ; 22 : 84 - 94 .
9. Hofer TP , Hayward RA . Are bad outcomes from questionable clinical decisions preventable medical errors? A case of cascade iatrogenesis . *Ann Intern Med* . 2002 ; 137 :327 - 333 .
10. Singh R, Singh A, Taylor JS, et al. Building learning practices with self-empowered teams for improving patient safety. J Health Manage 2006; 8: 91-118.
11. JCAHO . *Comprehensive Accreditation Manual for Hospitals: TheOff cial Handbook, Update 1* ; Oak Brook, IL: Joint Commission Resources ; 2001.
12. Marder R , Sheff RA . *The Step-by-Step Guide to Failure Modes and Effects Analysis* . Marblehead, MA: Opus Communications, Inc .; 2002 .
13. Singh R , Singh A , Fox CH , Taylor JS , Rosenthal TC , Singh G .Computer visualisation of patient safety in primary care: a systems approach adapted from management science and engineering . *Inf Prim Care* . 2005 ; 13 : 135 - 144.
14. Dovey SM , Meyers DS , Phillips RL Jr , et al . A preliminary taxonomy of medical errors in family practice . *Qual Saf Health Care* . 2002 ; 11 : 233 - 238.
15. Makeham MAB , Dovey SM , County M , Kidd MR . Ininternational taxonomy of errors in general practice: a pilot study. *Med J Aust* . 2002 ; 177 : 68 - 72.
16. Sheikh A , Hurwitz B . Setting up a database of medical error in general practice: conceptual and methodological considerations. *Br J Gen Prac* . 2001 ; 51 : 57 - 60 .
17. Bhasale A . The wrong diagnosis: identifying causes of potentially adverse events in general practice using incident monitoring. *Fam Pract* . 1998 ; 15 : 308 - 318 .
18. Britt H , Miller GC , Steven ID . Collecting data on potentially harmful events: a method of monitoring incidents in general practice . *Fam Pract* . 1997 ; 14 : 101 - 106 .
19. Fischer G , Fetters MD , Munro AP , Goldman ED . Adverse events in primary care identifi ed from a risk-management database . *J Fam Pract* . 1997 ; 45 : 40 - 46 .
20. Ely J , Levinson W , Elder NC , Mainous A , Vinson D . Perceived causes of family physicians errors . *J Fam Pract* . 1995 ; 40 : 337 - 344 .
21. von Neumann J , Morgenstern O . *Theory of Games and Economic Behavior* . 1953 ed . Princeton.
22. Plsek P. Redesigning health care with insights from the science of complex adaptive systems. In: Crossing the quality chasm: A new health system for the 21st century. Washington, DC: National Academies Press; 2001
23. Kotter J. Leading change. Cambridge, MA; Harvard Business School Press; 1996.
24. Wheatley M. Leadership and the new science: Learning about organizations from an orderly universe. San Francisco: Berret-Koehler Publishers; 1992.
25. Senge P. The fifth discipline: The art and practice of learning organization. New York: Doubleday; 1990.
26. Taylor FW. The principles of scientific management. New York: Harper and Brothers Publishers; 1911.

BIOGRAPHIES

Ranjit Singh MB BChir, MA (Cantab), MBA, MABFM, MD, a graduate of the University of Cambridge (England) where he read Medicine and Management. He is Associate Director and Professor, Patient Safety Research Center, State University of New York at Buffalo, USA. He has authored numerous papers on patient safety and simulation and served on the US National Expert panel on reporting in primary care

Chimay Anumba BScEng, PhD, DSc. He is a past Chair of the European Group for Intelligent Computing in Engineering and has been on the governing Council of the Institution of Civil Engineers of UK. He has authored more than 400 publications and has undertaken advisory/consultancy work for the UK Government and several firms. He is the Co-Editor of the Journal of Information Technology in Construction . He was awarded an Honorary Doctorate by Delft University of Technology in The Netherlands for outstanding scientific contributions to Building and Construction Engineering. He has won several awards, including an Engineering Foresight Award from the UK Royal Academy of Engineering. Currently he is the Head of Department of Architecture Engineering at the Pennsylvania State University. USA.

Ashok Singh, MB BChir, MA (Cantab), MABFM, MD, is a graduate of the University of Cambridge (England) where he read Medicine and Management. He is a Partner of The Niagara Family Medicine Associates, Niagara Falls. He has authored papers on patient safety and simulation

Gurdev Singh B.ScEngCivil (Aligarh), MScEng, PhD (Birm), is the founding Director of Patient Safety Research Center at State University of New York at Buffalo, USA. In 1996 he was recognized by the US Department of Justice as an 'Alien of Extraordinary Ability.' This classification is awarded to a person who has demonstrated that s/he has reached the top of her/his profession at the international level for a sustained period of time, or to a Nobel Laureate or equivalent. His experience and themes of his publications include: development and use of hybrid decision support systems, multidisciplinary appraisal of risk and reliability in projects and structures based on Monte Carlo simulation, multi-objective and multi-resource optimization and 'satisfisation', development of Singh-Markowitz efficient frontier method, concurrent engineering, stochastic and visual simulation of construction projects, and computer-aided learning of risk and reliability analysis

Acknowledgement

Authors are grateful for AHRQ R18 HS017020 Grant, the contribution of Ron Brooks, for writing the software and Diana Anderson for Project Coordination.

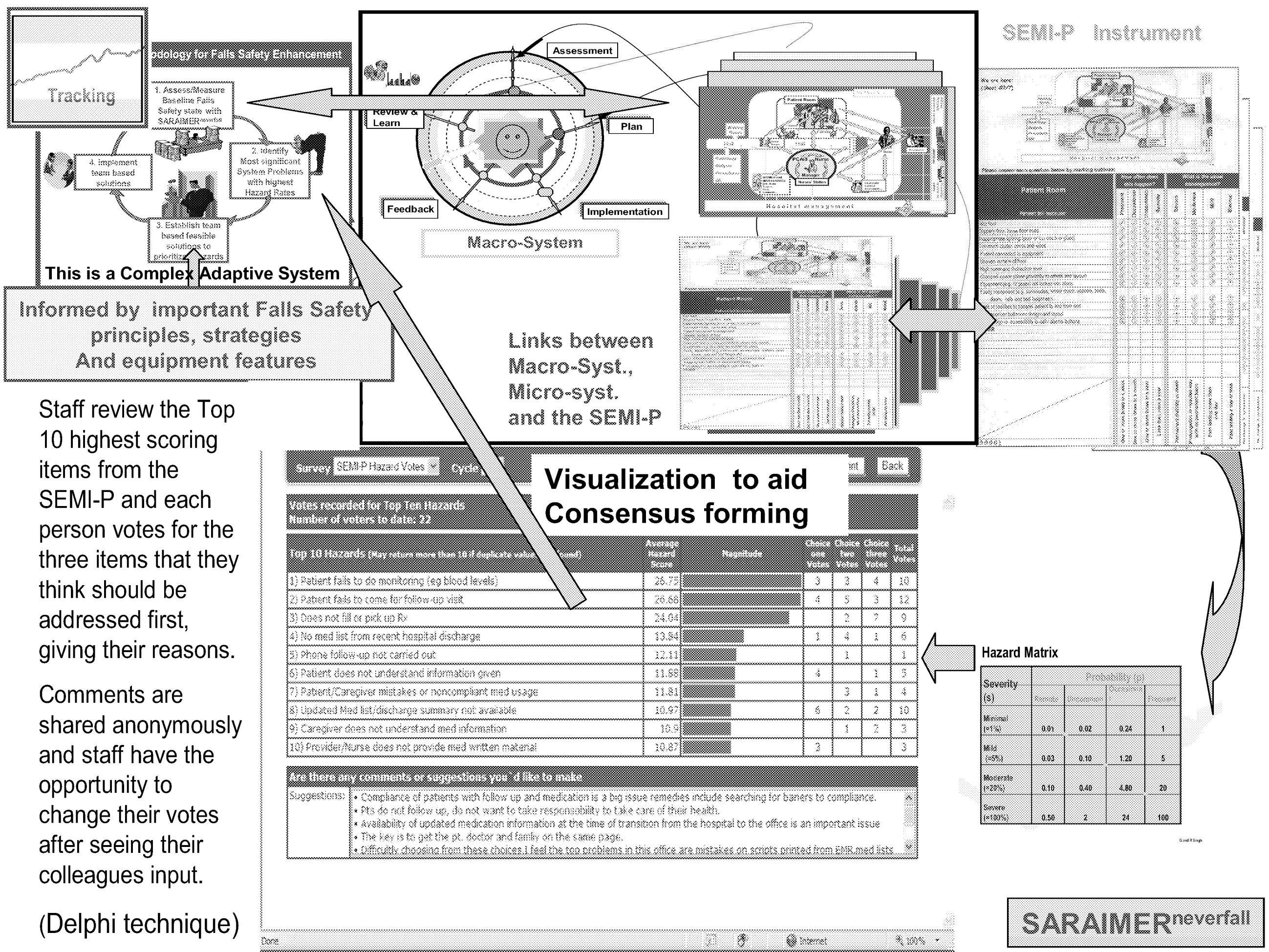


Figure 4: The Overall Schematic of the Systems Approach to Patient Safety Improvement. This illustration shows how the methodology can be adapted for any healthcare setting such as a hospital floor undertaking reduction of patient falls (**SARAIMER**[neverfall])

System Testing Using Use Cases for an Emergency Room Simulation Model

Guodong Shao
Manufacturing Simulation and Modeling Group
National Institute of Standards and Technology
Gaithersburg, MD 20899-8260, U.S.A.
E-mail: gshao@nist.gov

KEYWORDS
Modeling and Simulation, System Testing, Use Case.

ABSTRACT

Modeling and simulation (M&S) techniques are increasingly being used to solve problems and aid decision making in many different fields. It is particularly useful for Department of Homeland Security (DHS) applications because of its feature of non-destructive and non-invasive method of observing a system. Results of simulations are expected to provide reliable information for decision makers, but potential errors may be introduced in the M&S development lifecycle. It is critical to make sure to build the right model and that the model is built right. This paper identified the needs of system testing using specifications for M&S applications for DHS applications and providing a novel approach of Verification, Validation and Testing (VV&T) for DHS M&S community. System testing is an effective methodology that can help to ensure the functionality of a software system. It can also be applied to M&S applications. Use cases are usually used to specify requirements of a simulation system. The collection of use cases can cover the complete functionality of the simulation system and provide information necessary to generate test cases for system testing. Since use cases are associated with the front end of the M&S development lifecycle, testing can get started much earlier in the lifecycle, allowing simulation developers to identify and fix defects that would be very costly if found in the later stages. As an example application, a hospital emergency room (ER) simulation model was introduced. Use cases for the ER model were developed. Functional system test requirements and testing criteria of the ER model were discussed. Based on the coverage criteria, activity diagrams associated with the use case are created to capture scenarios and allow the specification of use case to be tested.

INTRODUCTION

Modeling and simulation (M&S) techniques are more and more being used to model real world problems in many different applications. M&S is an effective means to shorten real system development time by answering many what-if questions first. IEEE *standard glossary of* modeling and simulation terminology (IEEE 1989) has the definitions for model and simulation as "A model is an approximation, representation, or idealization of selected aspects of the structure, behavior, operation, or other characteristics of a real world process, concept, or system." "Simulation is a model that behaves or operates like a given system when provided a set of controlled inputs." M&S is the process of constructing a model of a system that contains a problem and conducting experiments with the model on a computer for a specific purpose of solving the problem and aiding in decision-making. The developers and users of the simulation models, the decision makers using the results of these models, and individuals affected by decisions based on such models are all concerned with whether a model and the simulation results are correct (Sargent 2007).

M&S is particularly valuable for DHS application, because M&S provides a non-destructive and non-invasive method of observing a system and also provides a way to test multiple inputs and evaluate various outputs (Donald and Brown 2005). Simulations allow users to reconstruct a comprehensive representation of real-world features during disaster response (Lisa 2006). The limitations of live exercises can be overcome through the use of simulation models that allow emergency response personnel across multiple levels in multiple agencies to be exposed to the same scenario. Simulation models can help the decision makers determine staff and resource levels in hypothetical terrorist attack scenarios (Shao and Lee 2007). These M&S applications often introduce new risks associated with potential errors in creating the model (programming errors) and inadequate fidelity (errors in accuracy when compared to real-world results). There are no established procedures for determining whether the results obtained from an M&S application are correct or satisfy real world needs. To ensure that a valid model and a credible simulation that produce correct results exist, verification, validation and testing of the model and the resulting simulation must be employed throughout the life cycle of an M&S application. (Cook and Skinner 2005). (Balci 2007) defines the model VV&T as follows: "Model validation is substantiating that the model, within its domain of applicability, behaves with satisfactory accuracy consistent with the study objectives. Model validation deals with building the right model. It is conducted by running the model under the same input condition that drive the system and by comparing model behavior with the system behavior. The comparison of model and system behaviors should not be made one output variable at a time. Model verification is substantiating that the model is transformed from one form into another, as intended, with sufficient accuracy. Model verification deals with building the model right. The accuracy of transforming a problem formulation into a model specification or the accuracy of converting a model representation in a micro flowchart into an executable computer program is evaluated in model verification. Model testing is demonstrating that inaccuracies exist or revealing the existence of errors in the model. In model testing, we subject the model to test data or test cases to see if it functions properly. "Test failed" implies the failure of the model, not the test. Testing is conducted to perform validation and verification. Some tests are devised to evaluate the behavioral accuracy (i.e., validity) of the model, and some tests are intended to judge the accuracy of model transformation from one forming another (verification)."

IEEE Standard Computer Dictionary (IEEE 1990) defines system testing as "System testing is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic." System testing is concerned with testing an entire system based on its specifications. It is independent of the process used to create any application. The tester evaluates the application from a user perspective. Internal design details are irrelevant and do not affect how tests are defined. The application's behavior, whether presented as use cases or other forms of requirements, drives the development of test cases (Tamres 2002). Effective system testing requires a concrete and testable system-level specification. A system specified with use cases provides much of the information necessary for system testing…the collection of use cases is the complete functionality of the system (Booch et al. 1999) (Kaner 2002). Unified Modeling Language (UML) (Fowler 2005) (Gomma 2003) use cases are usually used to define the M&S system requirement, specification and design.

Normally, testing addresses only verification by checking if the implementation meets the specifications. System testing using use case models also assists model validation. A complete analysis of the use case models not only evaluates whether the generated tests cover the requirements, but also evaluates whether the use case description meet the intended use needs (Hasling et al. 2008).

Traditionally, test case design techniques include analyzing the functional specifications, the software paths and the boundary values. These techniques are still valid, but use case testing provides a new perspective and identifies test cases in its unique way (Collard 1999). Early in the lifecycle of a software system there is no code to execute but there are models – requirement models, analysis models, architecture models, and others (McGregor 2007). Briand and Labiche presented the testing object-oriented systems with the UML functional system test methodology. They derive test requirements from use case description, interaction diagram (sequence or collaboration) associated with each use case, and class diagram (composed of application domain classes and their contracts). This early use of analysis artifacts is very important as it helps devising a system test plan, size the system test task, and plan appropriate resources early in the life cycle (Briand and Labiche 2001).

A test case is a description of a test with the expected outcome. A set of test cases can be created based on the use case of the simulation systems to verify if the model is correctly implemented according to its requirements. The test cases are defined as instantiations of the use cases of the simulation system. An important advantage of creating test cases from specifications is that they can be produced earlier in the development lifecycle and be ready for use before any codes are developed. Additionally, when the test cases are generated early, simulation developers can often find inconsistencies and ambiguities in the requirements specification and design documents. This will definitely bring down the cost of modeling and simulation systems as errors are eliminated early during the life cycle.

This paper describes a novel method to test DHS M&S applications. Currently there is no existing procedure within DHS for VV&T of M&S applications. Many DHS M&S applications are developed by different contractors. As a user, DHS may not be familiar with all the simulation tools and associated programming techniques, but they know what they want, understand the requirements well, therefore, performing a system testing using use cases to create test cases is a very useful approach to verify the requirements. Further more, the test cases can be reused if multiple contractors develop similar M&S applications using different tools.

As a DHS M&S application, a hospital ER simulation model is introduced to apply the system testing technique that generates test cases from use case specifications. The ER simulator is a discrete event simulation model of an emergency patient's flow in a hospital. The purpose of this simulation is to provide a small but realistic model of resources and patient's flow and congestion in the ER of the hospital in response to an emergency incident including the deployment of resources and actions for triage and treatment of the injured, movement of casualties to hospitals, and treatment at the hospitals. Ensuring the model's creditability is very critical. Only a correctly implemented model can provide valuable information for the hospital management teams to make the right decisions that will affect others including medical staff and patients. A system testing for the simulation model based on the UML use case model will assist to make sure the system meets the intended user needs and is implemented right.

This paper is organized as follows: the next section introduces a prototype of the hospital ER simulation model. Then use cases for the ER model are discussed. An example activity model is generated based on the use cases. Test requirements and criteria for the use case and the activity diagram are discussed. Finally the test cases associated with the use case and activity model are identified.

THE EMERGENCY ROOM SIMULATION MODEL

The emergency department simulator models the resources, patients flow and congestion in response to an emergency incident. The model demonstrates how the incident affects: dispatch of ambulance to transport of injured to the hospital, as well as the waiting time in different areas, and evaluates the resources needed according to different scenario. The simulation will allow hospital management teams to train by responding in real-time to crises that affect ER flow and evaluate the impact of their decisions.

The primary entities in the model are patients, medical records, and soiled linen; resources are medical staffs, and specialists, emergency vehicles, triage and exam rooms, test lab, and beds. Patients are modeled as first in – first out queues. The model allows the user to make modifications to selected model parameters through a graphical user interface. The user can change the number of patient arrival quantities and the average number of trauma and average number of cardiac patients per day. There are trauma rooms, cardiac rooms and specialty treatment rooms. Ambulatory and ambulance entrances exist as patient arrival points. The arrival of a cardiac or trauma patient, who will use more resources, will cause the backlog of regular patients (Shao and McLean 2008).

Model inputs

The inputs of the simulation model are listed as follows:
Patient's arrivals are modeled using statistical distributions.

- Number, location and type of casualties
- Availability of staff at work and off (on-call)
- Availability of resources
- Time and resources required for attending to each casualty type
- Probabilities of death from different casualty types over time.
- Hospital location
- Layout of the hospital
- Process stations
- Station capacities
- Processing times
- Patient arrivals rate
- Hospital shifts
- Medical resources
- Symptom-treatment profiles

Model outputs

The outputs of the simulation model may include the operation of the ER over time such as:

- System utilization
- Utilization of process stations and resources
- Updates of the status of the patients and medical staff
- Number of people treated and released, admitted, dead, waiting for treatment over time
- State of the staff and facilities (to determine their capability to deal with another incident)
- Run Time Interactions
- Simulated clock time – from Execution Control Supervisor
- Number of emergency medical technician*s* and ambulances dispatched over time to Traffic Simulation
- Number of ambulances and casualties arrivals over time from Traffic Simulation

Model logic

Figure 1 shows the model overview. There are two kinds of patients as arrival entities of the model: Ambulance and General. Ambulance patients are those patients who are in critical situation, such as trauma and cardiac patients. There are limited rooms and beds for ambulance patients. If all the rooms are occupied at the time; the patient has to be redirected to an alternate facility. After a patient is taken into the room, a Technician and Registered Nurse (RN) will treat the patient right away, create a medical record, and take the patient to the Medical Doctor (MD) for review. The MD will make a decision, and the patient will be moved to the nursing unit when the necessary procedures are done. General patients are ambulatory patients who can walk into the hospital and wait for an exam and treatment. They have to go through the triage process first. If all seats are taken, a triage-waiting area is provided. After the triage, patients are sent to the main waiting area waiting for calls to the different exam rooms based on their categories. Exam rooms include general exam, orthopedic exam, OB/Gyn exam, pediatric exam, and critical exam rooms. If it is not critical, the patient can be discharged. If further tests or X rays are needed, patients have to be in the queue for these procedures.

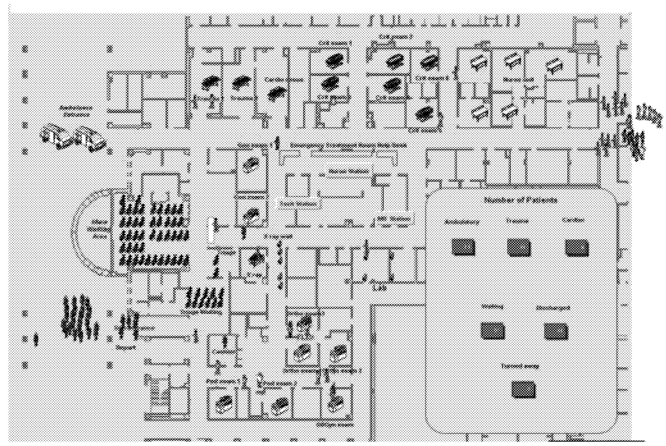


Figure 1: ER Model Overview

USE CASE

UML use cases are widely used to define the M&S application requirements. They are also used for the ER model. Use cases tell the user what to expect, the developer what to code, the technical writer what to document, and the tester what to test (McGregor 2007). They are used to describe sequences of actions that the simulation system performs as a result of input from the users; use cases help to express the workflow of the application. A use case describes interactions between users and system. This makes use cases independent from the implementation (EODiSP 2008) and reusable because they shall apply to every implementation of the system, regardless of what simulation tool is selected and the graphical user interface looks like.

Use cases represent the high level functionalities provided by the system to the user, so they are a good source for deriving system test requirements. When planning test cases for use cases, all possible execution sequences need to be identified and then covered during testing as they may be sources of different failures. But the use case diagram itself is not a very typical graph for testing; it is too high level and not many node and branches can be covered. However, a use case can be described in more detailed form as a table. The table provides details of operation and includes alternatives, which model choices or conditions during execution (Ammann and Offutt 2008).

Depicted in Figure 2 is the use case model diagram for the ER model. The ovals represent use case, and the stick figure represents actors that can be either humans or other software systems that interact with the simulation system. The lines represent communication between an actor and a use case. Each use case represents that functionality that is going to be implemented. In the context of the ER model, there are two kinds of actors (Shao and Lee 2007):

- *Simulation Analyst*: The *Simulation Analyst* is the core user of the system. The simulation analyst is responsible for executing the model and analyzing the simulation results on a daily basis. S/he might be involved in the simulation system development and is capable of performing data collection. The simulation analyst can define various scenarios for other users, verify the model based on the scenario, make suggestions regarding the length of the simulation run, the number of runs needed, and the initial conditions. S/he is responsible for analyzing the simulation results and documenting the findings.
- *Simulation User*: The *Simulation User* is the primary user of the system. By simulating different scenarios in a virtual environment using different settings, s/he is trained to respond to all kinds of situations. The response actions may include the deployment of resources, actions for triage, treatment of the injured, movement of casualties to other facilities, and transferring patients to another hospital/facility under different scenarios in the virtual world.

There are a total of 11 use cases in this use case model. The actor *Simulation Analyst* has five use cases; *define scenarios, initial/reset simulation, configure simulation environment, analyze simulation results,* and *turn-on facility layout*. The actor *Simulation User* has six use cases: *simulate patient arrival, simulate patient departure, simulate triage process, simulate emergency treatment, run simulation* and *simulate lab test and exam*.

As a sample, a detailed introduction of the *simulate patient arrival* use cases is provided in Table 1. The table will provide a basis for creating the activity diagram, which is more useful for testing.

ACTIVITY DIAGRAM

A UML activity diagram can be created based on a use case. An activity diagram shows the flow among activities. In many ways, UML activity diagrams are the object-oriented equivalent of flow charts and data flow diagram from structured development (Ambler 2004). Activities can be used to model a variety of things, including state changes, returning values, and computations. In this paper, the activity diagram is used to model the logic capture by the use cases as considering activities as user level steps. Two kinds of nodes are used: action states and sequential branches.

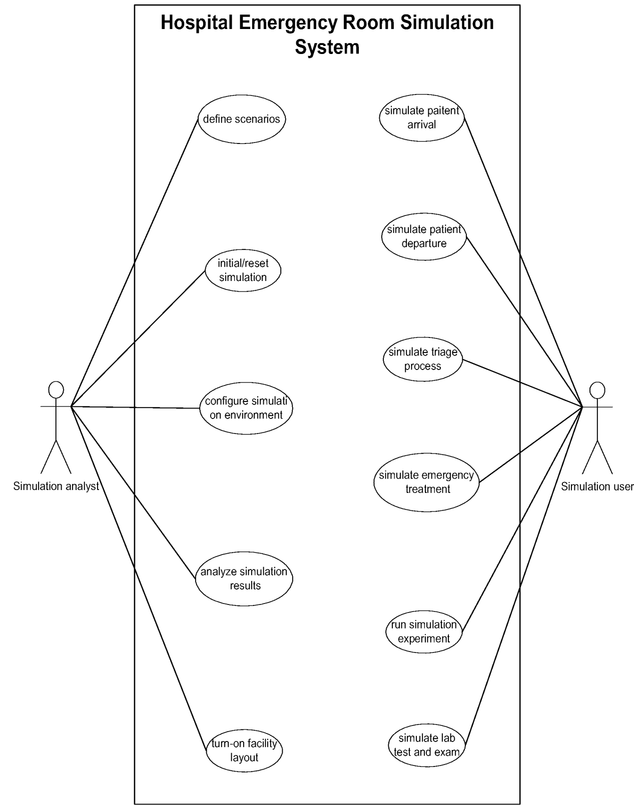


Figure 2: Use Case of ER Simulation System

Table 1: Use Case for Simulate Patient Arrival

Use Case Name	Simulate Patient Arrival
ID	1
Summary	Patient arrival rate and other characteristics are being entered and simulated
Actors	Simulation user
Preconditions	• Simulation software is launched • Simulation model is loaded • Simulation scenario is defined
Description	1 Simulation user starts to run the simulation model 2 Simulation system prompts user to select type of patient from a list (ambulatory patients, trauma patient, and cardiac patient) 3 Simulation user chooses the patient type 4 Simulation system prompts user to input number of patients 5 Simulation user inputs number of patients 6 Repeat step 2, 3, and 4 three times in order to enter all three kinds of patients 7 Simulation system executes with the patient type and arrival rate entered
Alternatives	If the user inputs invalid data, the simulation model will abort with an error message. Line 2, 3, 4, and 5: based on the different implementations, the user interface may vary, the way the user inputs data may be different.
Post conditions	• Patient type and quantities are entered into the system • Patient arrival rates are calculated and stored in the system • Simulation continues to run

The numeric items in the use case description presented in table 1 express steps that the actors undertake. These correspond to inputs to or outputs from the simulation model and appear as nodes in the activity diagram as action states. The alternatives in the use case represent decisions that the model or actors make and are represented as nodes in the activity diagram as sequential branches (Ammann and Offutt 2008). One activity diagram could represent several test cases because of decision points and data variations described in the activity diagrams.

The activity diagram for the "Simulate Patient Arrival" is shown in Figure 3. As described in section 2, there are three types of patients: General patients, trauma patients and cardiac patients. The user needs to input the number of patients for each type. Once all three types of patients are entered into the model, the system will check to see if the inputs are valid or not. If the input is valid, the simulation will continue to execute smoothly, otherwise, if any of the inputs is invalid, an error message will be displayed and the simulation will abort. In order to generate the test cases from the activity diagram that derived from the original use cases, we need to define the testing requirement and coverage criteria.

TESTING CRITERIA

There is no such thing as "complete testing" and "exhaustive testing." Coverage criteria are used to decide which test inputs to use and also provide useful rules for when to stop testing. The definition of test requirement and coverage criteria by (Ammann and Offut 2008) are: "Test Requirement: A test requirement is a specific element of a software artifact that a test case must satisfy or cover. Coverage Criteria: A coverage criterion is a rule or collection of rules that impose test requirements on a test set."

Ammann and Offutt introduced four distinct coverage criteria: Graphs, logical expressions, input space and syntax structures. In the use cases and activity diagram discussed in previous sections, where user language is used, there is no complicated predicate that contains multiple clauses, so logic coverage criteria is not useful. Also because there are no obvious data definition-use pairs, the data flow coverage criteria are not applicable. The two applicable criteria to use case graphs are node coverage and edge coverage. Test case values are derived from interpreting the nodes.

Another criterion for use case graphs is based on scenarios. A use case scenario is an instance of a use case, or a complete path through the use case. End users of the complete system can go down many paths as they execute the functionality specified in the use case. Multiple scenarios may be needed to completely describe a system.

Following the basic flow would be one scenario. Following the basic flow plus first alternate flow would be another. The basic flow plus second alternate flow would be a third, and so on (Zielczynski 2006).

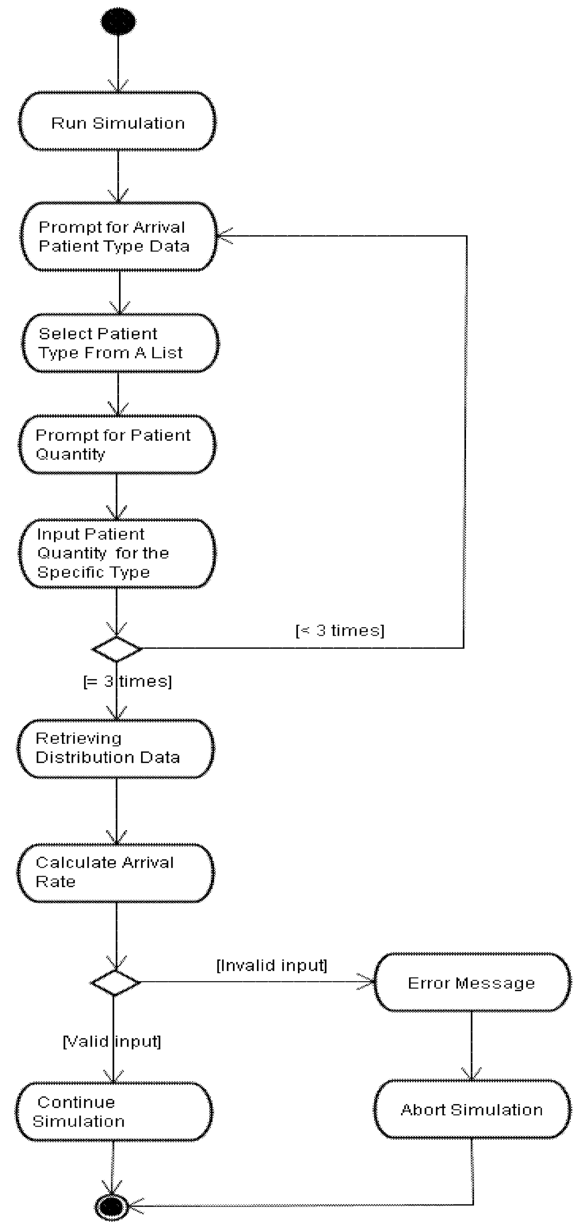


Figure 3: Activity Diagram for "Simulate Patient Arrival" Use Case

Figure 4 shows that every use case may have many scenarios; it is a one–to-many relationship. One scenario may also have many test cases, so it is also a one–to–many relationship. In this paper, we applied the scenario criteria to generate test cases for the ER model.

To create test cases from activity diagrams, every path or transition need to be considered. Test procedure for these test cases are used to verify successful and/or acceptable implementation of the simulation system requirements. This provides good traceability to original requirements, to test and verify requirements and to discover inconsistency in the requirements. Missing test cases are only a result of an incomplete use cases model (Hasling et al. 2008).

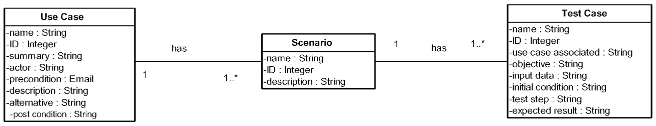


Figure 4: Relationship Diagram of Use Case, Scenario and Test Cases

TEST CASE

The purpose of a test case is to identify conditions that will be implemented in a test and expected results. Test cases are needed to verify acceptable implementation of the system requirement, which is a use case model in this paper. (Samurin 2008) defines test case as "a set of test inputs, executions, and expected results developed for a particular objective: to exercise a particular program path or verify compliance with a specific requirement."

An excellent test case should satisfy the following criteria (McGregor 2007):

- Reasonable probability of catching an error
- Exercises an area of interest
- Doesn't do unnecessary things
- Not redundant with other tests
- Makes failures obvious
- Allows isolation and identification of errors

Here is the three-step process for generating test cases from a fully detailed use case (Heumann 2001):

- For each use case, generate a full set of use case Scenarios such as a use case description table and activity diagrams.
- For each scenario, identify at least one test case (basic flow) and the conditions that will make it execute.
- For each test case, identify the data values that are used to test.

Based on the use case description, each combination of basic and alternate flows and the scenarios can be identified. Test cases can be created as soon as a use case is available, well before any code is written.

As an example, test cases for simulate patient arrival are created in Table 2 and Table 3. Table 2 presents the normal basic flow process, we need to make sure this scenario works correctly, and then we need to cover the major alternative path that the user can take through this use case and think about what could go wrong. Table 3 shows the invalid input scenario.

Data coverage for the test can also be specified. If you want to create tests with every possible data variation in every possible test path, you may end up with too many tests, impossible for you to handle. Therefore, sample the data variation choice in each test path is a practical way to do (Heumann 2001). We used the Input Domain Modeling (IDM) method discussed in (Ammann and Offutt 2008) and category - partitioning technique to decide the testing data values in the test steps. The details are not discussed in this paper.

Table 2: Test Case for "Simulate Patient Arrival" Use Case

Test Case Name	**Simulate Patient Arrival –Normal Basic flow process**
Use case name	Simulate Patient Arrival basic flow
Objective	To verify using valid patient arrival data
Input data	• Ambulatory patients: 250 • Trauma patient: 10 • Cardiac patient: 6
Initial conditions	1. The hospital ER simulation model is running. 2. Graphic user interface prompt for patient type selection
Test steps	1. Simulation user selects an unselected patient type from the list (ambulatory patients, trauma patient or cardiac patient). 2. Simulation System prompt "Enter the avg number of daily ambulatory patients (default avg =150);" "Enter the avg number of daily trauma patients (default avg=4);" or "Enter the avg number of daily trauma patients (default avg=4);" based on the patient 3. Simulation user enters a valid input (not one of the three 0, A, or 10000) 4. Repeat 1, 2, and 3 steps three times to cover all the three patient types. 5. Simulation system runs smoothly with the valid inputs entered.
Expected results	After the user input valid data, the simulation model will continue to run using the input to calculate patient arrival rate.

Table 3: Test Case for "Simulate Patient Arrival" Use Case

Test Case Name	**Simulate Patient Arrival – Enter invalid input**
Use case name	Simulate Patient Arrival alternate flow
Objective	To verify using invalid patient arrival data
Input data	• Ambulatory patients: 0 • Trauma patient: A • Cardiac patient: 200000
Initial conditions	• The hospital ER simulation model is running. • Graphic user interface prompt for patient type selection
Test steps	1. Simulation user selects an unselected patient type from the list (ambulatory patients, trauma patient, or cardiac patient). 2. Simulation System prompt "Enter the avg number of daily ambulatory patients (default avg =150);" "Enter the avg number of daily trauma patients (default avg =4);" or "Enter the avg number of daily trauma patients (default avg =4);" based on the patient types. 3. Simulation user enters an invalid input (one of the three 0, A, or 10000) 4. Repeat 1, 2, and 3 steps three times to cover all the three patient types.
Expected results	If any of the input is invalid, a error message will be displayed and simulation will abort.

CONCLUSION

M&S techniques are increasingly used to solve problems and aid decision making in many different fields, and are particularly useful for DHS applications because the actual system simulated may be impossible to be built, or has not been built yet, or testing an actual system is too dangerous or costly (Cook and Skinner 2005). Results of simulations are expected to provide reliable information for the decision makers to make wise decisions and predictions, but potential errors may be introduced in the process of the M&S development lifecycle. It is critical to build the right model and that the model is built right.

System testing is an effective methodology to help ensure the functionality of a software system. It can also apply to M&S applications. A well-defined concrete and testable system-level specification is needed for that purpose. Use cases are usually used to specify the requirements for a simulation system. The collection of use cases can cover the complete functionality of the simulation system and provide information necessary to generate test cases for system testing. Since use cases are associated with the front end of the M&S development lifecycle, testing can get started much earlier in the lifecycle, allowing simulation developers to identify and fix defects that would be very costly if found in the later stages. This also provides good traceability to original requirements, to test and verify requirements and to discover any inconsistency in requirements.

Using a use case model for test generation has been done in software development. This paper identified the importance of testing in early stages of the lifecycle of M&S, and presented the test methodology based on the UML use case diagram for DHS M&S applications. As a case study, a hospital emergency room (ER) simulation model was introduced. Use cases for the ER model were developed, and the use case description, activity diagram associated with the use case are created. Functional system test requirements and testing criteria of the ER model were discussed. We showed how activity diagrams can be used to capture scenarios and allow the specification of a use case to be tested. By executing the testing cases, we got expected results and improved the model based on the testing results. Problems such as array size and error messages have been fixed. The ER simulator is a relatively simple model; it's a good example to try out this system testing approach. This system testing approach can also be applied to more complex DHS or manufacturing simulation models.

This paper demonstrated a novel approach to test DHS M&S applications for the DHS community. Currently no procedure exists within DHS for VV&T of M&S applications. As a user, DHS may not familiar with all the simulation techniques, but understand the requirements well. Therefore, using use cases to create test cases is a very useful approach to verify the requirements. Further more, the test cases can be reused if multiple contractors develop similar M&S applications using different tools.

REFERENCES

Abdurazik, A. and J. Offutt. 2000. "Using UML Collaboration Diagrams for Static Checking and Test Generation". *In Proceedings of the Third International Conference on the Unified Modeling Language (UML '00),* York, UK, 383--395.

Ammann, P.; and J. Offutt. 2008. *Introduction to Software Testing*. Cambridge University Press, New York.

Balci, O. 2007. "Validation, Verification, and Testing Techniques throughout the Life Cycle of a Simulation Study." *Annals of Operation Research*, 53 (1994), 121-173.

Booch, G.; I. Jacobson; and J. Rumbaugh. 1999. *The Unified Modeling Language User Guide*. Addison-Wesley.

Briand, L. and Y. Labiche. 2001. "A UML-Based Approach to System Testing". *In Proceedings of the Fourth International Conference on the Unified Modeling Language (UML 2001)*. (Toronto, Canada, October 2001).

Collard, R. 1999. "Test Design: Developing Test Cases from Use Cases", *Better Software Magazine*. (Vol. 1, Issue 4). 1-10.

Cook, D. and J. Skinner. 2005. "How to Perform Credible Verification, Validation, and Accreditation for Modeling and Simulation." *The Journal of Defense Software Engineering.*

Fowler, M. 2005. *UML Distilled Third Edition A Brief Guide to the Standard Object Modeling Language.* Addison-Wesley, Boston.

Gomaa, H. 2003. *Designing Concurrent, Distributed, and Real-Time Applications with UML*, Addison-Wesley, Boston.

Heumann, J. 2001. "Generating Test Cases from Use Cases." *The Rational Edge*.

Kaner, C. 2002. "Black Box Software Testing – Professional seminar".

McGregor, J. 2007. "Test Early, Test Often." *Journal of Object Technology* (vol. 6, no. 4), 7-14. http://www.jot.fm/issues/issue_2007_05/column.

Sargent, R. 2007. **"**Verification and validation of simulation models**".** In Proceedings of the 2007 Winter Simulation Conference. IEEE, Picataway, N.J., 124--137.

Shao, G. and C. McLean. 2008. "Emergency Room Simulation Prototypes for Incident Management Training". *In Proceedings of Industrial Simulation Conference 2008*, (Lyon, France, June 2008). 323-327.

Shao, G. and Y. Lee. 2007. "Applying Software Product Line Technology to Simulation Modeling of Emergency Response Facility." *Journal of Defense Modeling and Simulation, The Society for Modeling and Simulation International*, Vol. 4(4), October 2007, http://www.scs.org/pubs/jdms/vol4num4/vol4num4.html.

Tamres, L. 2002. *Introducing Software Testing*, Addison-Wesley, Boston.

Zielczynski, P. 2006. "Traceability from Use Cases to Test Cases." *The Rational Edge*.

Hasling, B.; H. Goetz; and K. Beetz. 2008. "Model Based Testing of System Requirements Using UML Use Case Models". *In Proceedings of 2008 International Conference on Software Testing, Verification, and Validation.* 367--376.

EODiSP. 2008. "Use Cases and Test Cases." http://www.pnp-software.com/eodisp/resources/archive/useAndTest.

Samurin, A. 2008. "Test case template for those who are using or would like to implement the Use Case modeling technique." http://www.geocities.com/xtremetesting/TCtemplate.html.

IEEE. 1989. "IEEE Standard Glossary of modeling and simulation Terminology." IEEE STD 610.3.

IEEE. 1990. "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries." IEEE New York, NY.

Ambler, A. 2004. *The Object Primer: Agile Model-Driven Development with UML 2.0*. Cambridge University Press, New York.

Donald, R. and D. Brown. 2005. "Development of Metrics to Evaluate Effectiveness of Emergency Response Operations". *In Proceedings of the 10th International Command and Control Research and Technology Symposium.*

Lisa, P. 2006. "Simulation Model for Bioterronrism Preparedness in an Emergency Room". *In Proceedings of the 2006 Winter Simulation Conference.*

ENGINEERING SIMULATION

SIMULATION OF LONGITUDINAL GEOMETRIC YARN STRUCTURE

Ionut Neagu
Pascal Bruniaux
ENSAIT " Ecole Nationale Supérieure des Arts et Industries Textiles "
2, allée Louise et Victor Champier,
Roubaix, 59056, France.
ionut.neagu@ensait.fr

Dorin Avram
Vasile Airinei
Technical university "Gh. Asachi"
Str. D-trie Mangeron nr. 53, Corp TEX1, Iasi-700050, Romania
dorin.avram@tuiasi.ro
airineiv@yahoo.com

KEYWORDS
Yarn, simulation interfaces, model design, virtual reality, Computer Aided Design.

ABSTRACT

It is necessary to model the yarn before modeling the fabric in order to develop a three-dimensional environment for the virtual design and the simulation of clothing. Why this approach to the study of yarn geometry? The answers are simple enough. First of all, the industry is not completely pleased with actual simulators that achieve good overall performance but lack sensitivity and coherence. Secondly, the absence of a multilevel model that starts from fibers. That is why we must define a geometry adapted to the yarn and describe the behavior of the modeled yarn so we could integrate the yarn model into the fabric model.
The geometric complexity of textile structures and the anisotropy (non-linear behavior of fibers) represent the two main problems that prevent the simulation of interrelationship between fiber and yarn. The response of fibers to a mechanical load is influenced by their nature and interactions. Apart the above-mentioned observation, we also underline the major difficulty in describing the physical behavior of small fibers that often take an irregular form. Generally, the characterization of textile material follows a hierarchical strategy based on a multidimensional and structural concept leading to successive representations of geometrical models.
Most of the researchers have developed simulators directly from the yarn structure without taking into account the fiber structure and its characteristics. It is called geometric structure because the analysis of the yarn structure is based on geometric elements of yarns and fibers, such as: yarn diameter, position of fiber radius, pitch fiber angle, fiber length, length of a fiber element, surface and volume of fibers and yarns. It is known that fibers have a random position in the yarn which is manifested by a variable number of fibers in cross-section and different radial positions along the yarn. As a result, the description of the yarn structure is achieved through the analysis of the fiber position. There are many research papers which focus on the mechanic modeling of the geometry of the yarn structure but there is no model which takes into account the cause-effect of fiber integration in the yarn structure. Much effort can be saved if we avoid the trial and error approach and use instead, the simulation model. Even if there are many studies having as main theme the modeling and simulation of textiles, none of them focuses on the dynamic yarn behavior, i.e. none of them takes into consideration fiber characteristics when it comes to modeling the yarn structure.

THE GEOMETRICAL MODEL

We chose the idealized helical yarn geometry for the purpose of simulating the yarn structure. The main algorithm is described as follows:

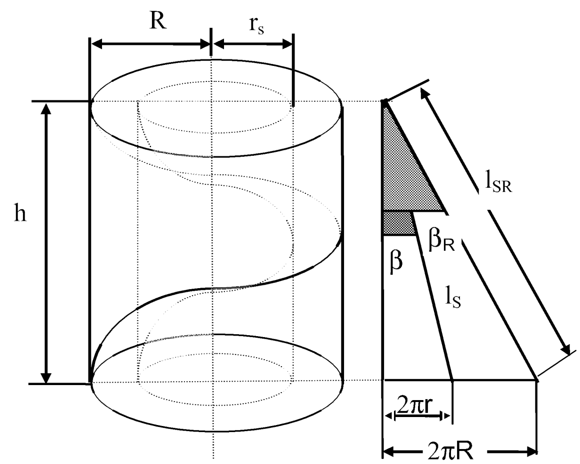


Figure 1: The idealized helical yarn geometry

Where:
R - Yarn radius;
r_S - Fiber radius;
β - Twist angle (angle between the fiber tangent and the yarn axis);
β_R - Fiber twist angle located on the yarn radius;
l_S - Length of the fiber helix, length corresponding to the layer radius;
l_{SR} - Helix length corresponding to the yarn radius;
h - Helix pitch.

The radial twist angle is determined by the relation:

$$tg\beta = \frac{2\pi r_s}{h} = \frac{2\pi r_s T}{1000} \qquad (1)$$

Where:
T - Yarn twists (twist/m).

The twists are determined by Köchlin's relation:

$$T = \alpha_m * \sqrt{N_m} \quad (2)$$

Where:
Nm - Average yarn count (m/g);
αm - Twist multiplier.

The helix pitch can also be calculated as:

$$h = \frac{T}{1000} \quad (3)$$

We calculate the yarn diameter by means of the following equation:

$$D = 2R = \sqrt{\frac{4}{Nm_F \cdot \pi \cdot \rho_F}} \quad (4)$$

Where:
D - Yarn diameter (mm);
ρF - Yarn density (g/cm^3);
NmF - Average yarn count (m/g).

The yarn density is determined as followed:

$$\rho_F = \nu \cdot \rho_f \quad (5)$$

Where:
ν - Packing rate is adopted between 0, 6 and 0, 7;
ρf - Average fiber density.

$$\nu = f(T) \quad (6)$$

In order to calculate the average fiber density, we use the following algorithm:

$$\overline{\rho_f} = \frac{4}{\pi \cdot Nmf \cdot d^2} \quad (7)$$

Where:
ρf - Average fiber density (g/cm^3);
Nmf - Average yarn count (m/g);
d - Fiber diameter (μm).

We calculate the mean fiber diameter as follows:

$$d = \sqrt{\frac{4 \cdot 1^6}{Nm_f \cdot \pi \cdot \rho_f}} \quad (8)$$

Where:
ρf - Average fiber density (g/cm^3);
Nmf - Average fiber count (m/g).

The number of fibers in a cross-section of the yarn can be obtained by:

$$n_s = \frac{Nm_f}{Nm_F} \quad (9)$$

The limit variation factor is determined by the relation:

$$CV_{\lim} = \frac{100}{\sqrt{n_s}} \quad (10)$$

Where:
ns- The number of fibers in yarn cross section of the yarn.

The mean square deviation is determined by the relation:

$$\sigma = \frac{CV_{ef} \cdot D}{100} \quad (11)$$

CVef - is the factor of effective variation of yarn diameter and is determined by the relation:

$$CV_{ef} = CV_{\lim} \cdot I \quad (12)$$

Where:
I - The irregularity factor that we adopt from the Uster statistics ranges between I = 1.2 ÷ 1.3.

The computation of diameter limits.

$$D_{\min} = D - 1{,}96 * \sigma$$
$$D_{\max} = D + 1{,}96 * \sigma \quad (13)$$

Where:
D - Average yarn diameter;
σ - Standard deviation of yarn diameter.

Computation of the twist angle

The twist angle and the fiber trajectory are calculated using the result of in-plane development of the helix portion between sections (M) and (N). The result is a right triangle ABC as shown in figure 2, whose sides are AB = 1 and AC = rφ.
The angle (β), formed between the tangent to the helix and the direction of the yarn axis, is called twist angle.

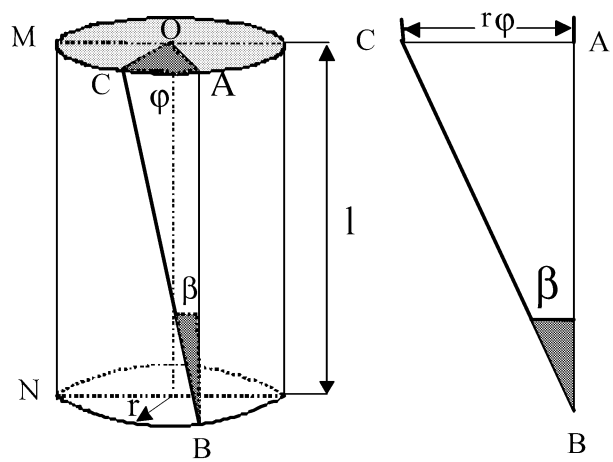


Figure 2: The position of the fiber within the yarn structure.

It results:

$$tg\beta = \frac{r\varphi}{l} \quad (14)$$

Where: φ represents the rotation angle (expressed in radians) of section M with respect to section N. If we represent the rotation angle in terms of the number of complete rotations made by section M with respect to N, then:

$$tg\beta = \frac{2\pi nr}{l} \quad (15)$$

and since the ratio n/1 represents the twists, then:

$$tg\beta = 2\pi r T \quad (16)$$

Where:

β - Twists angle (expressed in degrees);
r - Fiber radius (mm);
T- Yarn twists (twists/mm).

This relation is very important for the tubular structure of yarns and represents the fundamental twist formula. It brings together the main parameters of the tubular structure of yarns.
The result is that for a yarn with tubular structure whose twist is (T), the twist angle will be wider if the fibers are on the outer boundary of the yarn.
-for fibers at yarn core level: r = 0 and β = 0.
-for fibers on the outer boundary of the yarn:
r = R and β = βR.
Then:

$$tg\beta_R = 2\pi RT \qquad (17)$$

Where:
R - The yarn radius (mm);
T - Yarn twists (twists/mm).

Calculating the Fiber Length

As a result of the twisting process, fibers take the shape of cylindrical helixes, except for the central fiber which remains straight.
There is a cylindrical helice situated on a cylinder whose radius is r, as shown in figure 3. On the helix we consider the point M which is characterized by:
- φ - The angle of helix AOM';
- r - Helix radius, OM'=r.

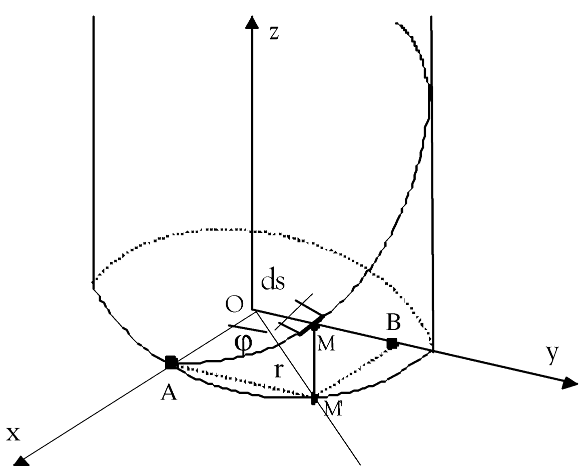


Figure 3: Calculating the length of the helical arc

The parametric equations of point M are:

$$\begin{aligned} x &= OA = r \cdot \cos\varphi \\ y &= Ob = \sin\varphi \\ z &= MM' = k \cdot \varphi \end{aligned} \qquad (18)$$

Where:
k - Represents the helix rate and is equal to the ratio between *dz* and *dφ:*

$$k = \frac{dz}{\varphi} \quad (19)$$

Then:

$$dz = k \cdot d\varphi \qquad (20)$$

and through integration we have:

$$z = k\varphi + c \qquad (21)$$

Out of these conditions:

$$\begin{aligned} &\varphi = o \Rightarrow z = 0 \;\; soc = 0 \\ &\varphi = 2\pi \;\; z = h \end{aligned} \qquad (22)$$

It results:

$$k = \frac{h}{2\pi} \quad (23)$$

Where h is the helix pitch.

We consider the helix length ds as shown in figure 2.
The length ds is written as:

$$ds = \sqrt{dx^2 + dy^2 + (dz)^2} \quad (24)$$

By replacing dx, dy and dz, we have:

$$ds = \sqrt{(r^2 \sin^2\varphi + r^2\cos^2\varphi + \frac{h^2}{4\pi^2})d\varphi^2} \qquad (25)$$

From the previous computation, we have the relation:

$$ds = d\varphi\sqrt{(r^2 + \frac{h^2}{4\pi^2})} = rd\varphi\sqrt{1 + \frac{h}{4\pi^2 r^2}} = rd\varphi\sqrt{1 + \frac{1}{4\pi^2 r^2 T^2}} \qquad (26)$$

For a helix rotation angle equal to 2π, we obtain the length of loop (corresponding to a helix pitch) s, situated at a distance r from the center of the cylinder.
It results:

$$s = 2\pi r\sqrt{1 + \frac{1}{tg^2\beta}} \qquad (27)$$

By replacing, $4\pi^2 r^2 T^2 = tg^2\beta$ the relation can be rewritten as follows:

$$s = \frac{2\pi r}{\sin\beta} \qquad (28)$$

Thus, the fiber length corresponding to a pitch of the right cylindrical helix represents the loop length s, which depends on the radial position of the fiber r and the twist angle of the fiber.

SIMULATION RESULTS

We have developed a software program in Visual C++ language, based on the previously presented algorithm. We used a multiple-document interface for the dialog windows and OpenGL API to represent the main 3D window with the 3D structure of the yarn. In figure 4 we can see the main panel, with the input data at the top and the output data at the bottom. The modification of input data determines the automatic computation of output data, due to an Update Data function. The 3D viewer window is presented in figure 5; on the upper-left side, we have the saving options of the structure, and also a slider for changing the twist angle. The yarn section is rendered by open packing distribution, taking into account the average number of fibers; the core is represented by a straighten fiber around which the other fibers are disposed in layers. There are 5 layers in all, but in figure 5 only 3 are shown, for purely visual reasons. Every fiber layer has a different color in order to be easily identified. Table 1 indicates how we calculate the layer output.

Table 1: Layer Output

Layer number	Layer radius	ns / layer	∑m*
1	r_{s3}	m*1	m*1
2	r_{s2}	m*2	m*1+m*2
3	r_{s3}	m*3	m*1+m*2+m*3
			∑m *=ns

Where:
n_s - The number of fibers in yarn cross-section;
r_s - Layer radius.

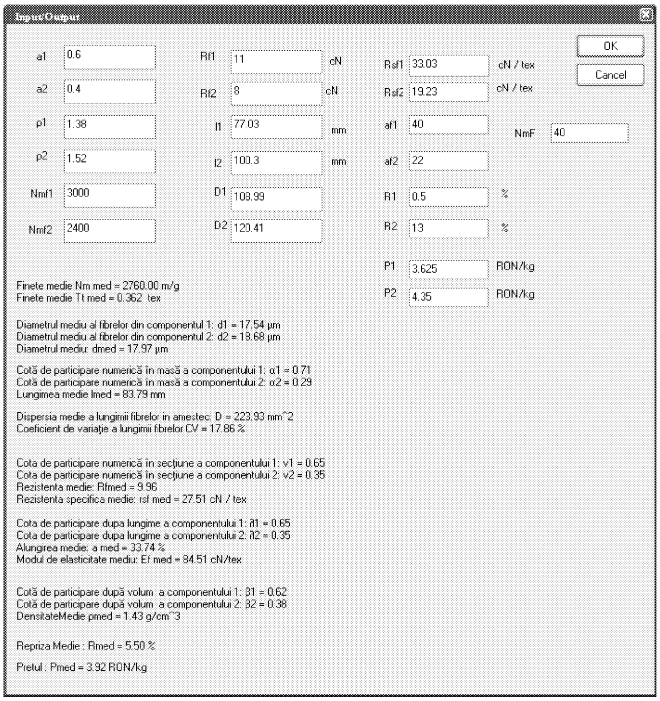


Figure 4: The main panel with input and output data

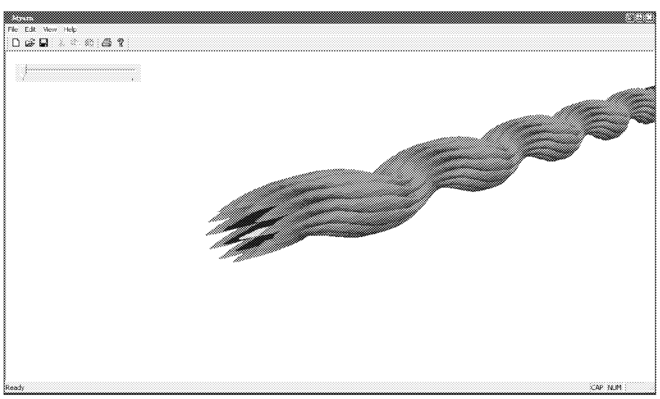


Figure 5: The 3d view panel with the yarn structure

The input data including yarn and fiber characteristics are indicated in table 2, and the output data, in table 3.

Table 2: Input Data

Input data	Not.	Value	UM.
Average yarn count;	NmF	39.50	m/g
Average fiber count;	Nmf	1792	m/g
Packing rate;	N	0.50	-
Yarn density;	ρF	0.76	g/cm^3
Fiber density;	ρfb	1.31	g/cm^3
Yarn torsion;	T	820	t. /m
Irregularity index;	I	1.20	-
Yarn diameter.	Dmed	0.287	mm

Table 3: Output Data

Output data	Not.	Value	UM.
Yarn minimum diameter	Dmin	0.27	mm
Calculated yarn diameter;	Dmedc	0.31	mm
Yarn maximum diameter;	Dmax	0.34	m/g
Fiber average diameter;	Dfmed	0.05	mm
Number of fibers in the yarn cross-section;	ns	45	-
Yarn diameter coefficient of variation;	CvD	17.82	%
Limit coefficient of variation;	CVlim	15	%
Effective coefficient of variation;	Cvef	18	%
Standard deviation of yarn diameter.	σ	51	-

* Observation: The difference between the average yarn diameter obtained by means of microscopic measurements and the calculated diameter is of 22%, which verifies Barella's study.

CONCLUSIONS

Although there are many studies which discuss the modeling and simulation of textiles, none of them focuses on the 3D structure of the yarn based on fiber characteristics, i.e. none of them takes into consideration fiber characteristics when it comes to modeling the yarn structure. The yarn length is hard coded in the source code itself; in the future we intend to change this as input data. This time we avoided this operation for statistical distribution purposes.
All research work must be continued so it could serve industrial development.

REFERENCES

Avram Dorin, 2004 "Yarns' Structure", Iasi, ISBN 973-7994-46-9,
Bruniaux P., X.Legrand, C.Ngo Ngoc & J.M.Castelain 2002 "Modélisation mésoscopique du tissu par le fil ", XIIth Romanian Conference of Textiles and Leadership, Iasi, Roumanie`.
Bruniaux P., X.Legrand & C.Vasseur, 2006a, "Yarn and fabric model inter-connected", Journal of Advanced Materials, Vol. 37, N°4, pp 60-69.
Bruniaux P. & C.Ngo Ngoc, 2006b, "Nonlinear model of a fabric warp and weft", Advances in Complex Systems, Vol. 9, N°1-2, pp 99-120.
Djaja, R. G., Moss, P. J, Carr, A. J., 1992, "Finite Element Modeling for an Oriented Assembly of Continuous Fibers", Textile Research Journal, Vol. 62, No. 8, 1992, pp 445-457

Grishanov, S. A., Lomov, S. V., Cassidy, T. Harwood, R. J., 1997, "The Simulation of the Geometry of a two component yarn Part II: Fiber distribution in the yarn cross-section", Journal of textile Institute", Vol. 88, Part 1, No. 4, 1997, pp 352-372.

Grishanov, S. A., Harwood, R. J. Bradshaw, 1999, "A model fiber migration in staple-fiber yarn", Journal of textile Institute, 90 Part 1, No. 3, 1999, p 298-321.

Ghith A., Bruniaux P. & Vasseur C. 1997 "Modeling fabric dynamics", Ith Romanian Conference of Textiles and Leather, Iasi, Roumanie, 23-24 October.

Jiang, Y; Chen, X., 2005, "Geometric and algebraic algorithms for modeling yarn in woven fabrics", Journal of Textile Institute, Vol. 96, 2005, pp 237-245.

Keefe, M., Edwards D. and Yang, J., 1992 "Solid Modeling of Yarn and Fiber Assemblies", Journal of the Textile Institute, Vol. 83, No. 2, 1992, pp 185-196.

Keefe, M., 1994, "Solid Modeling of Fibrous Assemblies: Part I, Twisted Yarns, Journal of the Textile Institute", Vol. 85, No. 3, 1994, pp 338-349.

Langenhove, L. A, 1997 "Simulating the Mechanical Properties of a yarn Based on the Properties and Arrangement of its fibers Part I: The Finite Element Model", Textile Research Journal, Vol. 67, No. 4, 1997, pp 263-268.

Legrand X., P.Bruniaux & C.Vasseur, 2000a "A new approach for modeling woven fabrics", WAC'2000, Fourth Biannual World Automation Congress, 9 pages, Wailea, Maui, Hawaii, U.S.A.

Legrand X., P.Bruniaux & C.Ngo Ngoc, 2000b, "Modeling and Study of the Dynamical Behavior of a Yarn", 16th IMACS World Congress 2000, Lausanne, Suisse, 7 pages, 21-25 August, (2000).

Ngo Ngoc C., P.Bruniaux & J.M.Castelain , 2002a, "Modeling friction for yarn fabric simulation, Application to bending hysteresis", Congress ESS2002, 14th European Simulation Symposium and Exhibition, Simulation in Industry, pages. 433-436, Dresden, Germany, 23-26 October.

Ngo Ngoc C., P.Bruniaux & J.M.Castelain, 2002b, "Constrained dynamic yarn modeling", Textile Research Journal, Vol. 72, N°11, pp 1002-1008

Ngo Ngoc C., 2003, "Modélisation non linéaire et simulation des matériaux souples textiles. Application aux essais Kawabata ", Thèse de doctorat en Automatique, Université de Lille Flandres Artois

Octav Brudaru, Copaceanu C, 1999 "Genetic Algorithms and Hybrid Methods for the Assembly Line Balancing", International Textile, Clothing and Design conference.

MODEL OF THE CONDENSATE WATER AND THE AIR COOLED CONDENSER OF A COMBINED CYCLE POWER PLANT

Yadira **Mendoza-Alegría**
Edgardo J. **Roldán-Villasana**
Instituto de Investigaciones Eléctricas,
Gerencia de Simulación
A.P. 475, 62000 Cuernavaca, Morelos , México

Rafael **Cruz-Cruz**
Comisión Federal de Electricidad
Centro Nacional de Capacitación y
Adiestramiento de Operadores Ixtapantongo,
Valle de Bravo, Edo. de México

email: eroldan@iie.org.mx URL: www.iie.org.mx

KEYWORDS

Combined Cycle Power Plant, Real Time Simulator, Process Modeling, Air Cooled Condenser.

ABSTRACT

In this paper, the basis of the models of the condensate water and the air cooled condenser are presented. The models are part of a full scope simulator of a 450 MW combined cycle power plant. The simulator is executed in real time and is intended to be a support for the training of the operators of the Comisión Federal de Electricidad (the Mexican utility company). The simulator is presently in the final acceptance tests stage and is programmed to be in commercial operation in 2010. Here, they are included a summary the modeling methodology used to develop the referred models and the mathematical fundaments used to obtain the main equations. The tendencies of selected variables during a transient are displayed and analyzed in order to probe the validity of the new generic models.

INTRODUCTION

The Simulation Department (GS) of the Electrical Research Institute (IIE) has by now about thirty years of experience in the development of real time simulators for operators' training. The GS is also dedicated to computer based training systems and process emulators for testing and adjustment of real control systems. Note that some acronyms are defined after the name spelled in Spanish.

The main costumer of the GS is the Comisión Federal de Electricidad (CFE) in particular, the National Center for Training and Teaching of Operators Ixtapantongo (CNCAOI). Presently, this center has five full scope simulators for power generating plants, three classroom simulators, and one more portable that travels to all the generation centers for training in specific maneuvers.

The GS has also worked for the Geothermal Simulation Center of the CFE developing three simulators (including one classroom's). The simulator of the Unit 1 of the Nucleoelectrical power plant that was in service since 1990 to 2005 was developed by the GS too.

In all cases, the GS offer maintenance services in all of its products. Every product is developed considering the particular needs and characteristics of every customer.

During 2007, the GS finished the development of a full scope simulator for a gas turbine power plant (Roldán-Villasana et. al, 2008). This simulator was delivered to the CFE and it is currently in commercial operation in the CNCAOI. Based in this work, a simulator for combined cycle power plant (SCC) was developed based on Unit II of the plant Chihuahua that nominally generates 450 MW (Zabre et al., 2009). The plant has two gas turbines (with a generator each); two heat recovery steam generator (HRSG) units, one steam turbine, one condensate system, one feedwater system and diverse the auxiliary systems. The simulator has a full scope and is a replica but the control system of the gas turbine that is Mitsubishi in the real plant and Siemens in the simulator. The second gas turbine is simulated in a simplified way.

COMBINED CYCLE POWER PLANT SIMULATOR

The SCC has a full scope and is able to perform several functions, as: Run/Freeze; Backtrack; Variant Simulation Speed; Establishment of Initial Conditions; Snapshooting; Malfunctions; Changes of External Parameters; Remote Actions; Actions Registration; Several Development Tools (useful during the development and testing of a new model e.g. monitoring and tabulation of variables in real time).

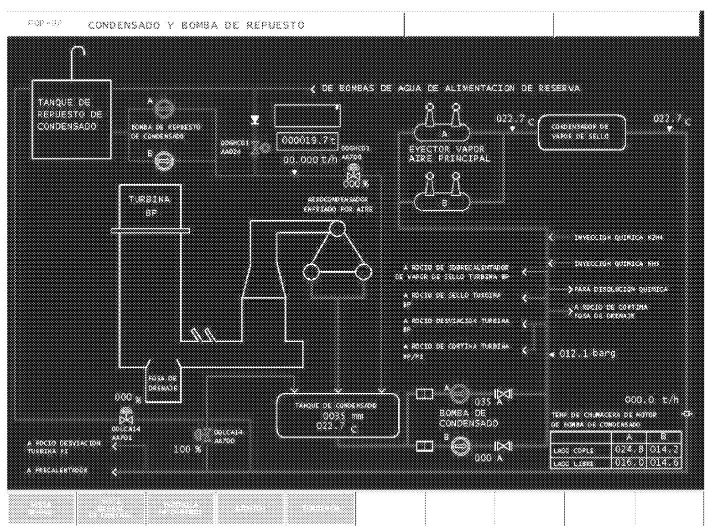


Figure 1: Control Screen for the Condensate Water System

The simulator is controlled by an instructor from its own console. The operators are working in control stations, replica of the real ones. An example of a control screen from the simulator is presented in Figure 1 for a part of the condensate systems and makeup pumps.

The systems included in the simulator are presented in Table 1.

Table 1: SCC Systems

Included Systems
Gas Turbine: Full Gas Turbine Plant with 13 (including the electric system and generator) systems and its respective controls; a second Gas Turbine Plant simplified.
Water: Condensate Water, including the ACC, Feedwater System.
Steam: HRSG1; HRSG2, Steam of High, Intermediate and Low Pressure.
Turbine: Turbines of High, Intermediate and Low Pressure; Metals of Turbines (temperatures, vibrations and eccentricities).
Electric System and Generator: Electric System and generators of the Steam Unit; Primary and Secondary Regulations.
Auxiliary: Auxiliary Steam; Gland Turbine Steam; Cooling Water System; Lubrication Oil; Control Oil; Hydrogen for the Generator; Efficiency Reports.
Minimized: Instruments and Services Air; Chemical Dosage; Drains Systems (Water, Steam; common and Chemicals); Chemical Analysis; Potable Water; Services Water; Demineralized Water Plant; Sanitary Waste Water; Local Controls.

Modeling Methodology

The methodology used to develop the simulator may be summarized in the next nine points (Roldán-Villasana et al., 2009) considering a capacitive and resistive nodes approach (Colonna and Van Putten, 2007):

a) Getting the design basis information; b) Justified simplification of the systems in a diagram with the nomenclature to be used in all documentation; c) Designing the configuration of the flows and pressures network and parameterization of all the elements (valves, pumps, fittings) at the nominal operation point in an automated excel file that allows its importation into the simulator data base; d) Programming of the energy balance using the appropriate generic models; e) Programming the capacitive nodes like tanks, headers, condensers, heaters, etc., using the appropriate generic model; f) Local test execution; g) Integration of the different simulated systems (including the controls); h) Fabric test application and solution of discrepancies; i) Final acceptance test application and solution of discrepancies.

The methodology has been applied to develop models for operators' training into a real time simulation environment. In this case it is necessary to consider simplifications of the real process and assumptions for the modeling, for example: lump several pipelines in parallel into one equivalent, consider thermodynamic equilibrium for air-vapor-water phenomena, use of ideal gas in particular conditions, etc.

A more complex application of this methodology could be for equipment and process design. There are no reasons to doubt that the methodology may be apply for this kind of problem, but the generic models, the mathematical tools, and the simulation environment should be adapted. Anyway the IIE has not yet used it for design purposes.

Modeling Tools

Basically the modeling tools are a set of generic programs that may be personalized to any structure of the modeled object. All of them are proprietary of the IIE: Simulation Control and Interface (MAS); Generic Model to Simulate Flows and Pressures in Hydraulic Networks (*flupre*) (Mendoza et al., 2004); Generic Model to Simulate Electrical Networks (*redele*) (Roldán-Villasana and Mendoza-Alegría, 2004); Generic Model to Simulate Direct Contact Condensers with Non-Condensable Gases (*gecoin*); Generic Model to Simulate a Combustion Chamber (*gecomb*); Thermodynamic Properties for Water and Steam and for Mixtures of Hydrocarbon Components; Generic models to Simulate Particular Equipments (Open Tanks to Atmosphere, Pressurized Tanks, Air or Gas Containments, Electrical Response of Induction Motors, Condensers, Heat Transfer Equipment); PID Control Modules; Energy (Temperature or Enthalpy) Balance in Mixing Nodes; Mixing of Streams with Hydrocarbon Components; Mathematical Packages to Solve Linear and Non-Linear Algebraic Equations Systems; and Methods for Numerical Integration of Differential Equations.

SYSTEM DESCRIPTION

The condensate system is formed by the ACC, the condensate tank, the condensate pumps, the makeup tank and pumps, the drain system and the demineralized water tank and pumps.

Process Description

The condensate water system initiates from the low pressure turbine discharge up to the deaereator level control valve.

The main task of the system is to condensate the steam from the turbine via the fans of the ACC to be stored in the condensate tank from where it is distributed, through the condensate pumps, to several services of the plant mainly toward the feed water system. The air concentration in the cycle is maintained low thanks to the vacuum system that suctions the air from the ACC with the ejectors. During the trajectory, the water cools the air ejectors, condensate the turbine gland steam and is pre-heated if the plant is working with gas fuel (the pre-heater is not used if the plant works with Diesel). In any case the warming of the condensate water keeps the good efficiency of the cycle.

The condensate water is used too to seal valves, to feed the auxiliary boiler, the protection fire tank, to makeup the closed circuit for cooling, the filling lines of the HRSG, and the attemperators of the turbine seals and turbine by passes.

The makeup system keeps the level of the condensate tank through pumping from the makeup tank which is fed from the demineralized plant.

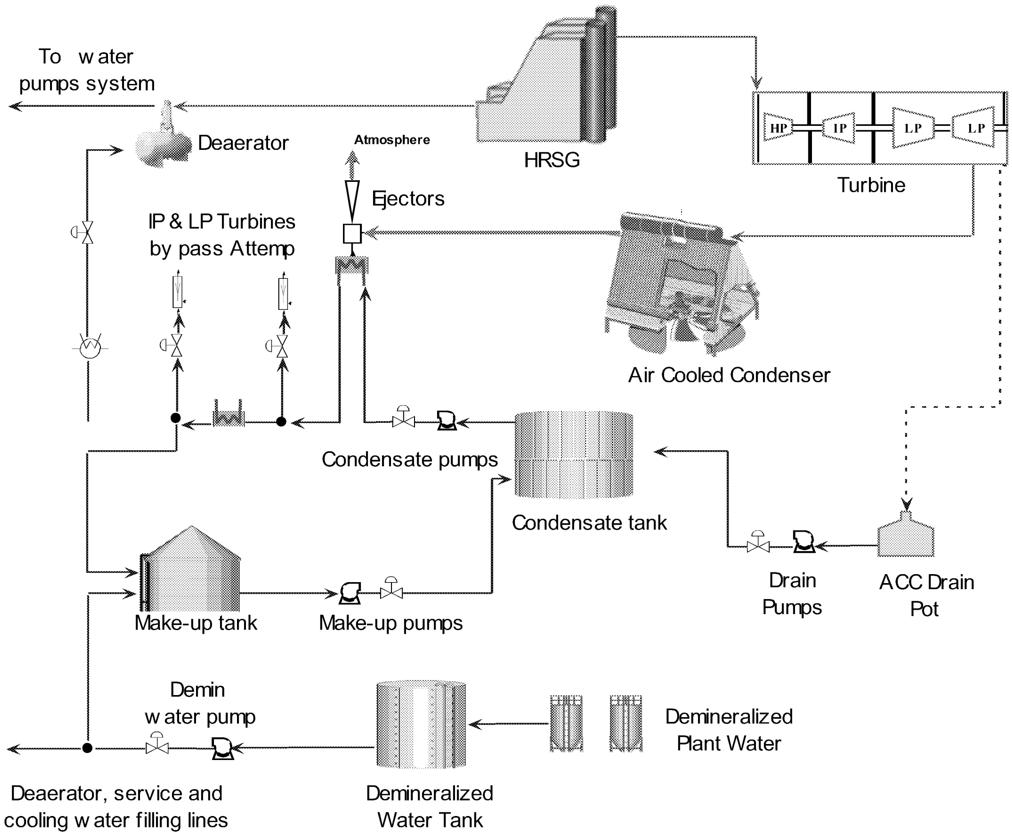


Figure 2: Simplified Diagram of the Air Cooled Condenser and Condensate Water System

Figure 2 is a simplified diagram of the main parts of the ACC and condensate water system, including the connections with other systems.

Air Cooled Condenser

The ACC is dry a heat exchanger (there is no contact between the steam and the cooling air). During the normal operation, it condensates the steam from the low turbine escape (which contains a small fraction of air). The condensation is carried out by the action of 30 fans that forces air flow through the condensers. The steam is distributed through a header towards finned tubes (primary condenser where it is partially condensed (up to twenty five fans). The water is drained to the condensate receptor and the steam (with air) is forced trough the reflux condenser (with five fans) where the water, by gravity, returns to the condensate receptor by slipping in the inner walls of the tubes, while the air is suctioned by the ejectors. From the condensate receptor the condensate is sent to the condensate tank. In Figure 3, a simplified representation of the ACC is presented.

MODELS DESCRIPTION

The models were designed and programmed by applying the methodology developed by the IIE that includes some modeling tools. In this section a summary of the principles of modeling and the final equations for the main components of the system are presented.

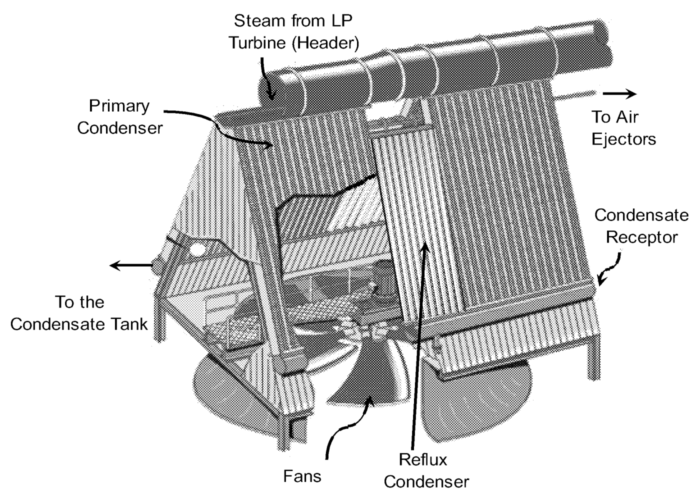


Figure 3. Air Cooled Condenser

Resistive Nodes: Flows and Pressures Networks

The *flupre* model is a generic program to solve any flows and pressures networks developed by the IIE. The model is derived from the momentum equation (in its integrated form) applied in each stream and the continuity equation for each node. For example, for pumps it may be assumed:

$$\Delta P = K_1^{'}\, w^2 + K_2^{'}\, w\,\omega + K_3^{'}\, \omega^2 - \rho\, g\, H \qquad (1)$$

where P is the pressure, K'_i are characteristic constants of the pumps(to adjust the performance curve), w is the mass flowrate, ω is the speed, ρ is the density, g is the gravity acceleration and H is the difference in height.

For non rotating elements (any flow restrictor) the equation is:

$$w^2 = K' \rho \, Ap^{\gamma} \left(\Delta P + \rho g H \right) \tag{2}$$

where the aperture Ap applies only for valves, but may represent a variable resistance factor to the flow (for example when a filter is getting dirty). The exponent γ of the aperture represents the characteristic behavior of a valve. Thus, in a given moment density and speed have a defined value and any element may be represented by a curve plotted in a ΔP vs w graph. For example, for the case of a pump (but the same result may be obtained for any other element), a curve of flow w on the x axis and ΔP on the y axis may be represented as in Figure 4.

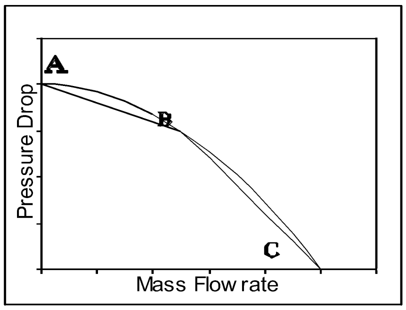


Figure 4. Flow vs Pressure Drop Representation

In the curve two straight lines may represent an approximation of the curve. In this case two straight lines are used to simplify the explanation, but the model allows for any number of them. For a given flow w, the pressure drop may be approximated by the correspondent straight line (between two limit flows of this line). If there are two or more elements connected in series an equivalent equation may be stated. In the same way, the elements connected in parallel between two nodes may be "grouped" to obtain an equivalent expression for the stream. The resultant expression may be arranged as:

$$w = K_1 \, \Delta P + K_2 \tag{3}$$

With this expression for each stream and the mass balance on the nodes, a linear equations system is obtained where pressures are the unknowns. Flows are calculated by the last equation once the equations for pressures were solved. Thus, the flow and pressures systems are based on algebraic equations, considering that the forces are balanced instantly before any changes in the independent conditions (external pressures, density, angular speeds, aperture of valves, etc.).

Flupre also includes an algorithm based on compact matrices to detect the topology that has to be solved in each integration step (Roldán-Villasana and Cásares, 1995). Some streams and its associated variables are eventually eliminated depending on the topology.

In Figure 5, the *flupre* configuration of the condensate tank and pumps, that is a part of the condensate water system which is composed, in its flows and pressures part, by three more *flupre* networks: demineralized system, makeup system and drain pot system.

Nodes, Tanks and Condenser (Capacitive Nodes)

There are two kinds of flowrates junction nodes.

The first one is a node that is part of the flows and pressures network whose pressure is calculated by *flupre*. In this case, a second state variable, the enthalpy h, is necessary in order to determinate all the variables of the node:

$$\frac{dh}{dt} = \frac{\sum w_i \, (h_i - h) - q_{atm}}{m} \tag{4}$$

In this equation, m is the mass of the node and q_{atm} the heat lost to the atmosphere. The subindex i represent the inlet conditions of the different flowstreams converging to the node. With the enthalpy and pressure it is possible to verify if the node is a single or a two phase one. The state variable could be the temperature if no phase change is expected and if the heat capacitive is constant and it divides the q_{atm} term.

The second kind of node is those than is a frontier of the flows and pressures network. This situation applies only for steam or two phases nodes. For the steam, these joints are used as capacitive nodes where the inertial effects are concentrated. The pressure is calculated with basis in an ideal gas behavior:

$$\frac{dP}{dt} = \frac{RT}{V}\left(\frac{dn}{dt}\right) + \frac{Rn}{CpV}\left(\frac{dh}{dt}\right) \tag{5}$$

The change in the moles n of the nodes is obtained correcting the change of the mass (being this a state variable too) with the molecular weight.

For this system, only two kinds of tanks were necessary to be simulated. Both of them are open to the atmosphere. For the first one it is not necessary to consider changes in its temperature and it is not required an energy balance and only the changes in the water level N is calculated. For example, for a vertical cylindrical tank (with a transversal area A_{tk} constant), the appropriate equation is:

$$\frac{dN}{dt} = \frac{\sum w_i - \sum w_o}{\rho \, A_{tk}} \tag{6}$$

If the transversal area is not constant should be necessary to consider the functionality between the area and the liquid level in the equation.

For the open tanks where the changes in temperature are necessary to know, the eq. (4) applies and a variation of eq. (5) is used being the mass the state variable.

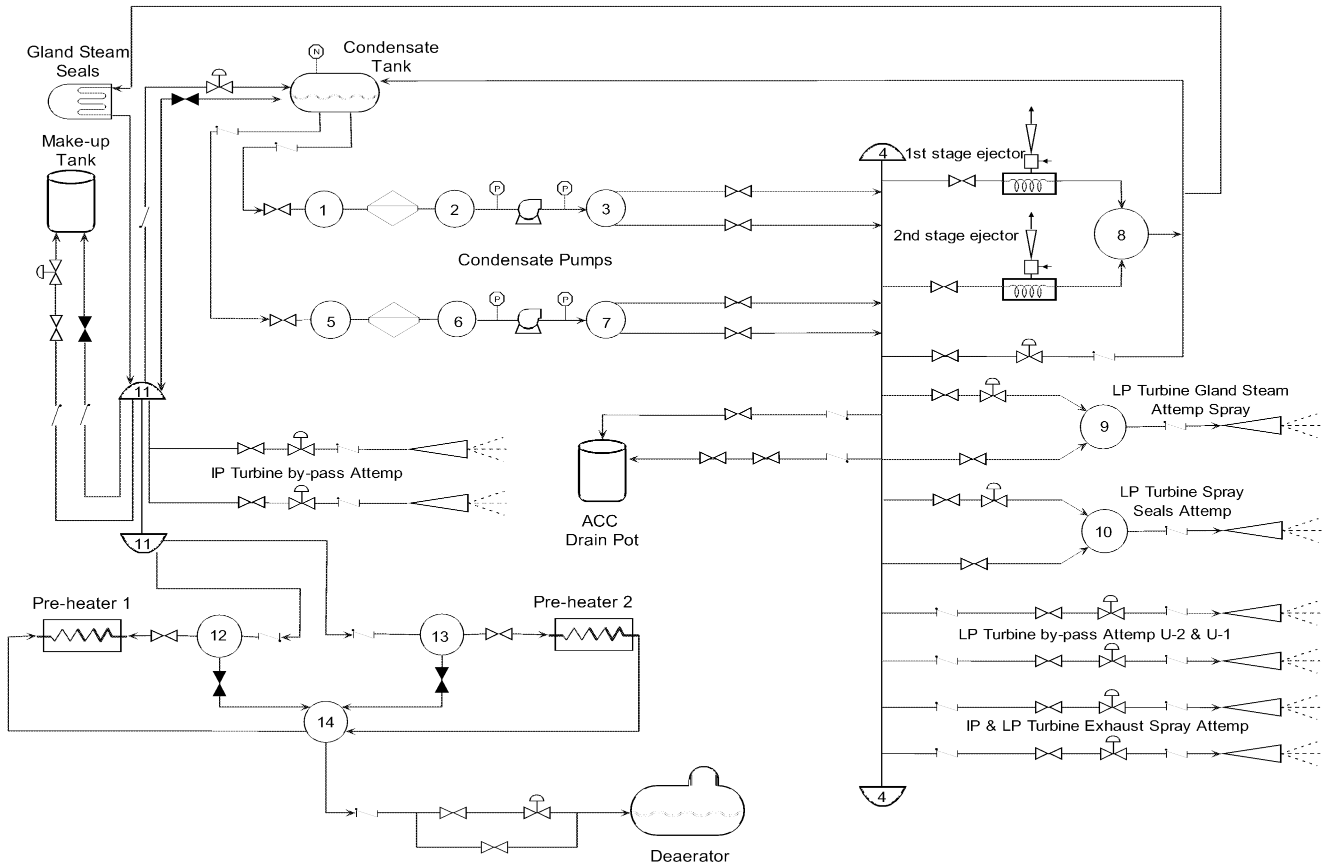


Figure 5: The *Flupre* Configuration Network, Sección of the Condensate Tank

The condensate tank was modeled as a closed recipient where a thermodynamic equilibrium is considered to exist at any moment between the liquid and the vapor. Non condensable elements are included (air) and the condensation and evaporation phenomena are implicitly simulated. For the gas mixture, the Dalton's Law is applied and the air behaves like an ideal gas (Roldán-Villasana and Vázquez, 2010). The main equations are represented as follow:

$$\frac{dm_l}{dt} = f\left(m_l, v, \frac{dv}{dP_v}, \frac{dP_v}{dt}, \frac{dm_t}{dt}, \frac{dm_a}{dt} \right) \quad (7)$$

$$\frac{dP_v}{dt} = f\left(prop, m, M, P_v, Cp_a, v, \frac{dv}{dP_v}, \frac{dm_t}{dt}, \frac{dm_a}{dt}, \frac{dprop}{dP_v} \right) \quad (8)$$

Where: v is for both saturated liquid and steam, m_l is the liquid mass, m represents the mass of all the components, M is the molecular weight of the species, CP is the heat capacity and *prop* is any thermodynamic property (enthalpy, density and temperature).

The derivatives of the thermodynamic properties in saturated conditions are based on a single variable of state, in this case of the vapor pressure:

$$\frac{dprop}{dt} = \frac{dprop}{dP}\frac{dP}{dt} \quad (9)$$

Air Cooled Condenser

The ACC uses a generic model to calculate the exit temperatures of the equipment (air and steam/water/air) and the phase (quality of the steam/water/air stream). A summary of the equations is presented here.

Figure 6 presents a schematic diagram of the ACC as was modeled. The main simplifications are to consider that: each condenser (primary and reflux) are represented by one heat exchanger each; the water condensed in the reflux condenser does not goes down due to gravity but it is separated from the air in a virtual node after the reflux condenser; the flows (cooling air and steam) works in a countercurrent way; and that the split factor of the collector node is constant.

For each stage of condensation, the interchanged heat q is a function of the global coefficient of heat exchange U, the transference area A and the mean logarithmic temperature difference ΔT. The value of $U\,A$ depends on the construction parameters of the equipment, the conditions of the fluids and the mass flow rates of streams, cold (c) and hot (h).

$$q = U\,A\Delta T \quad (10)$$

$$\Delta T = \frac{\Delta T_c - \Delta T_h}{\ln(\Delta T_c / \Delta T_h)} \quad (11)$$

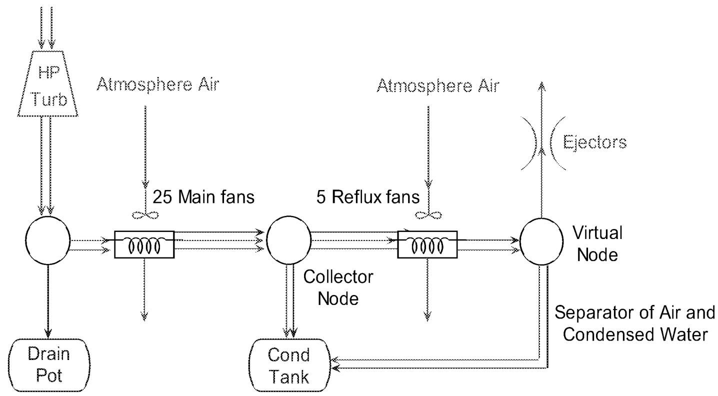


Figure 6. Schematic Diagram of the ACC Model

The mass flow rate of the air/steam mixture is calculated normally like sonic flow by the *flupre* generic model. The air flowrate depends on the number of fans turned on ad of their speed. To calculate ΔT, the outlet temperatures are initialized with values of the last integration time. The model may work even in the case the steam would be cooler than the atmospheric air (during a start-up and with a high concentration of air).

Establishing an energy balance for the air, assuming a heat capacity to be a constant:

$$T_o = T_i + \left(\frac{q}{w_a \, Cp} \right) \tag{12}$$

For the steam, the balance is, using enthalpies:

$$h_o = h_i + \left(\frac{q}{w_v} \right) \tag{13}$$

It is necessary to accomplish the thermodynamics' second law by avoiding the crossing of the temperatures in the ACC, considering a minimal temperature differences (defined by the user) between the hot and cold streams: the outlet temperature of the cold stream does not be warmer than the inlet temperature of the hot stream and the outlet temperature of the hot stream does not be colder than the inlet temperature of the cold stream, even considering the condensation of the steam. To this, the outlet temperatures are calculated with equations (12) and (13), the latter case verifying if the steam has been condensed and calculating the vapor quality (the steam is assumed to have always the pressure of the condensate tank). So, saturated, superheated and subcooled properties could be used, depending on the current conditions of the ACC.

If a crossing of temperatures is detected, clearly the heat calculated by eq. (10) is too high and the limiting stream is identified (depending of which is the cold and the hot streams and the maximum heat allowed by each stream to no violate the second thermodynamics law. So, the heat is calculated using the equation (12) or (13). With the new heat, the outlet temperatures are calculated again. An iterative method is necessary for this procedure.

The air and steam flowrates that flow to the condensate tank and the ejectors depends on a split factor empirically adjusted as a function of the condensate tank pressure and the total flow through the reflux condenser.

RESULTS

May be the more difficult part of the simulator's development is the validation of the models; to make it easier, local tests are performed in order to detect abnormalities or unexpected behavior in the isolated models. Once the simulator is integrated many scenarios should be verified (the simulator is intended to represent the plant in all the possible operational states), so a very detailed operation procedures ("Acceptance Simulator Test Procedures") elaborated by the client is used to corroborate the proper simulator´s performance.

An important part of the simulator development is the test to validate the behavior of the models. The tests are designed and executed by the costumer and according with the results the simulator may be eventually accepted. The tests included to probe the simulator in all the operation range of the plant, from cold start up to full range operation and cold down procedures. The response of the models are also verified under the presence of malfunctions (non programmed events that alters the normal operation like the trip of pumps, closure of valves, rupture of pipelines, fouling of heat exchangers, etc. The simulator accomplishes the norm *ISA-S77.20-1993 Fossil-Fuel Power Plant Simulators Functional Requirements*.

The test presented in this paper is not part of the official test acceptance procedures and was executed with the entire simulator coupled in closed circuit. No corrective action by the operator was made.

The transient consists in testing the model under severe malfunctions conditions: the trip of twelve fans followed by a leaking of the vacuum breaker valve allowing the entrance of air to the condensate tank.

In Table 2, the events presented in the test are summarized.

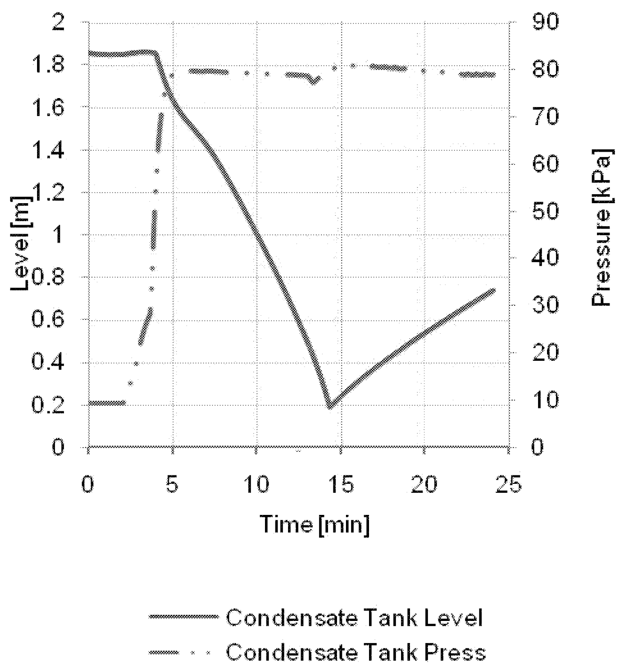
Table 2. Time and Programmed Events of the Test Transient

Time [min]	Event
0	Initiate the simulation. Steady state condition at 100% of load. 20 primary fans of the air condenser and 5 fans of the reflux ACC are in service.
2	Initiate sequential trip of 12 fans, 8 of the primary ACC and 4 of the reflux ACC.
3.5	Initiate malfunction of leaking in the vacuum breaker valve. Air goes into the condensate tank.
24	End of the test.

Graphic 1 shows the behavior of the pressure and level of the condensate tank and in the Graphic 2 the behavior of the temperatures in the condensate tank and in the inlet and exit of the ACC and the mass of the air in the condensate tank.

When the twelve fans trip, both, the pressure in the condenser and the temperature increase due the loose of cooling. The temperature of the reflux ACC (with only one fan in service), tends toward the value of the temperature of the primary ACC. At this point the level in the condenser does not change because there are still governed by its control. The air mass in the condensate tank remains unchanged.

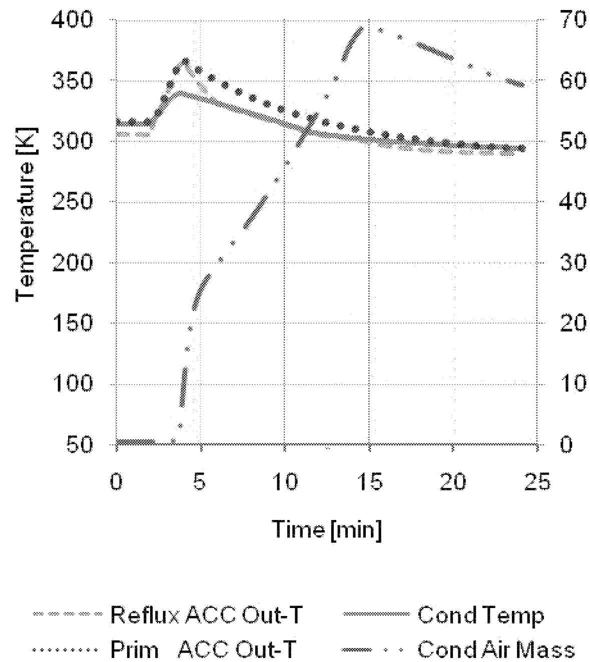
The loss of seal in the vacuum breaker valve malfunction is activated and there is a change in the rate of increase the pressure due the augmentation of the air mass in the condensate tank. The level is still controlled. The condenser temperature decreases due to the entry of cold air. Temperatures at the exit of ACC continue to rise. The air mass in the condenser increases due to the malfunction.



Graphic 1. Pressure and Level in the Condensate Tank

At the fourth minute the unit trips due a low vacuum in the condensate tank. The mass in the exit headers of the ACCs are bottled and temperatures drop due to local condensation. At almost six minutes, the temperature of the reflux ACC reaches the saturation point (condensate tank temperature) and it remains unchanged due to the phase change. At 15 minutes vapor condenses in the head completely and the liquid tends down to atmospheric temperature. A similar phenomenon occurs in the output header of the primary ACC. The condensate tank pressure increases to a maximum, following the temperature then descends a little below the atmospheric pressure due to the suction of the ejector. Therefore, the air mass continues to increase until it reaches a maximum and then decreases slightly.

The level decreases due to the trip of the unit and that trend continues until the condensate pumps trip because low level at about 16 min when starts a climb due the flowrate from the makeup pumps (which are small and are not able to compete with the condensate pumps).



Graphic 2. Air Mass and Temperatures

CONCLUSIONS

The presented results (and those obtained in the acceptance tests) demonstrate that the simulator represents the reference plat in an adequate manner and it is precise enough to be a viable and valuable tool for the combined cycle power plants operator's training.

The robustness of the models is due, in an important part, to the generic models developed by the IIE and the methodology that supports the development of the simulator. The Simulation Control and Interface, designed for both, the development and the commercial exploitation stages, are an important sustain of this methodology.

The combined cycle power plant have demonstrated to work in a more efficiently way that the conventional plants, are cheaper (and faster) and to construct and may be operated in a more easy way. These characteristics indicate that more plants of this type are going to be constructed in the near future. The CNCAOI has now a powerful tool to support the operator´s training of this kind of plants.

ACKNOWLEDGMENT

This work was carried out in the project: Development of a Simulator for Combined Cycle and the authors thank the staff of CNCAOI for their support in the implementation of activities and to Dr. Eduardo Gleason, manager of the GS for its contribution for a successful conclusion of the project.

REFERENCES

Colonna P., van Putten H., "Dynamic Modeling of Steam Power Cycles. Part I - Modeling Paradigm and Validation", *Applied Thermal Engineering*, 27, 467-480, 2007.

Mendoza Y., Roldán-Villasana E.J., Galindo I., Romero G., "Methodology to adapt the feedwater and condensate system, using flow and pressure generic model, for the Laguna Verde nuclear power plant simulator", *Summer Computer Simulation Conference*, The Society for Modeling and Simulation International, San Jose California, July, 2004.

Roldán-Villasana E.J., Cásares M.E., "A Generic Model to Solve Flow and Pressure Hydraulic Networks", *Proceedings of the Summer Computer Simulation Conference*, Ottawa, Canada, July, 1995.

Roldán-Villasana E.J., Mendoza-Alegría Y., "Adaptation of a Generic Model to Solve Flows and Pressures Hydraulic Networks for the Solution of Electrical Network", *Proceedings of CNS 6th International Conference on Simulation Methods in Nuclear Engineering*, Canadian Nuclear Society, Montréal Québec, Canada, October, 2004.

Roldán-Villasana E.J., Mendoza-Alegría Y., Zorrilla-Arena S. J.J., Cardoso G. M.J., Cruz-Cruz R., "Development of a Gas Turbine Full Scope Simulator for Operators' Training", *European Modelling Symposium, Second UKSim European Symposium on Computer Modelling and Simulation*, UK Simulation Society, 6-8 September, Liverpool, England, 2008.

Roldán-Villasana E.J., Cardoso M.J., Mendoza-Alegría Y., "Modeling Methodology for Operator's Training Full Scope Simulators Applied on Models of a Gas-Turbine Power Plant", *9° Congreso Interamericano de Computación Aplicada a la Industria de Procesos CAIP'2009*, Montevideo, Uruguay, Agosto, 2009.

Roldán-Villasana E. J., Vázquez A. K. *"Model of the Deaerator and the Feedwater System Implanted in a Combined Cycle Power Plant Simulator for Operators´ Training"*, To be publish in 2010

Zabre E., Roldán-Villasana E.J., Romero-Jiménez G., Cruz R., "Combined Cycle Power Plant Simulator for Operator's Training", Second International Conference on Energy and Sustainability 2009 Bologna, Italy, 2009.

BIOGRAPHIES

Yadira Mendoza-Alegría received her B.Sc. in Chemical Engineering from the Autonomous University of the State of Morelos in 1991. Since 1992 she is a researcher of the Simulation Department of the Electrical Research Institute. She has been involved in the developing of several real-time simulators for operators' training, mostly for conventional, nuclear, geothermal and combined cycle power plants. Her interest areas are modeling and simulation of thermo-hydraulic phenomena and heat and mass transfer process.

Edgardo J. Roldán-Villasana got his BSc degree as Chemical Engineer from the Autonomous Metropolitan University (México) in 1980; his MSc degree in the Process Specialty (Chem. Eng.) from the National Autonomous University of México in 1986; and his PhD in Process Simulation from the University of Manchester, Institute of Science and Technology (UK) in 1992. Since 1980 is a researcher of the Simulation Department of the Electrical Research Institute. He is head of projects and his work is related with the areas of modelling and simulation of thermal-hydraulics processes and transport phenomena. He has published about 40 papers in international journals and congresses, including the chapter in a book.

Rafael Cruz-Cruz He got his BSc in Electronics and Communication Engineering from the Instituto Politécnico Nacional of México in 1982 and his Master in Administration and High Direction from the Universidad Autónoma de Coahuila in México. Since 1983 he works in the National Centre for Operators' Instruction and Training Ixtapantongo of the Comisión Federal de Electricidad (the federal utility company in México) as head of the maintenance and intendancy department. He coordinates the projects related with training simulators of generation units.

MODELS OF THE TURBINE GLAND AND AUXILIARY STEAM SYSTEMS FOR A FULL SCOPE SIMULATOR OF A COMBINED CYCLE POWER PLANT

Edgardo J. **Roldán-Villasana**
Ma. de Jesús **Cardoso G**.
Adriana **Verduzco B**.
Jorge **Zorrilla A**.
Instituto de Investigaciones Eléctricas,
Gerencia de Simulación
A.P. 475, 62000 Cuernavaca, Morelos , México

Rafael **Cruz-Cruz**
Comisión Federal de Electricidad
Centro Nacional de Capacitación y Adiestramiento
de Operadores Ixtapantongo
Valle de Bravo, Edo. de México

email: eroldan@iie.org.mx URL: www.iie.org.mx

KEYWORDS
Power Plant Simulator, Process Modeling, Steam Models, Combined Cycle Power Plant.

ABSTRACT

In this paper the general description of a full scope simulator for a combined cycle power plant is presented; the antecedents of this work are explained; the basis of the models of the auxiliary and turbine gland steam systems are exposed and some tests are reproduced in order to the demonstrate the validity of the models. The simulator reproduces the operational behavior of the power plant "Chihuahua" located in Chihuahua, México.

INTRODUCTION

Since 1980, the Electrical Research Institute (IIE) (note that some acronyms are written after their name in Spanish) has developed several Power Plant Training Simulators, one for a partial scope portable simulator for turbine heating and rolling operations; two for conventional power plants; one portable for turbine start-up operations including the hard control panels; one for the Mexico City Metropolitan Subway System; two for Combined Cycle Power Plants; one for the Laguna Verde Nuclear Power Plant; one for a Dual (Carbon/Fuel) Power Plant; and one for a Gas Turbine Power Plant. This late simulator (STG) was finished in 2007 for the Mexican Utility Company (CFE), specifically for its Ixtapantongo National Centre for Operator´s Training and Teaching (CNCAOI). In 2008 the CFE asked at the IIE to expand the simulator to consider the steam cycle to conform a combined cycle power plant simulator (SCC).

The simulator was based on the Combined Cycle Power Plant Chihuahua II, located on the north México. This plant generates 450 MW (roughly 150 MW for each of the three electrical generators, two moved by gas turbines and one by a steam turbine) and started its operation on May 2001. The SCC has one steam turbine, two gas turbines, and two Heat Recovery Steam Generators (HRSG). One of the gas turbines is actually the same STG developed previously; the second one is a simplified gas turbine with a reduced scope, this means, that it has dynamic values of the main processes variables, adjusted to obtain the necessary thermal power for the steam turbine.

The simulator is intended to train operators. Although the development of training simulator is no longer a novelty, this is now the best option to exit the situation of power facilities suffering from a shortage of skilled workers (Molchanov et al. 2008).

The simulator is embedded in a system that allows controlling the simulation scenarios via the activation of malfunctions; run/freeze states; simulation speed; one step simulation; edition of initial conditions; tabulation; selection of instructors; backtrack; automatic scenarios; automatic snapshoting; external parameters; monitoring; actions recording; and environmental noises, between others (Zabre et al. 2009).

The simulator is presently in the final tests stage.

GENERAL DESCRIPTION OF THE SIMULATOR

Hardware Architecture

The architecture of the SCC is presented in Figure 1. The architecture is constituted by four Personal Computers interconnected through a fast Ethernet Local Area Network. The Instructor Console or simulation node (NS) PC has two 20" flat panel monitors, the Operators PCs (E01 and E02) have each also two 20" flat panel monitors, but the E01 has two additional 42" flat panel monitors. Each of these PCs has a Intel Pentium D processor with 3.6 GHz, 1GB of RAM, 74.4 GB HD, and Windows XP SP2 as operating system.

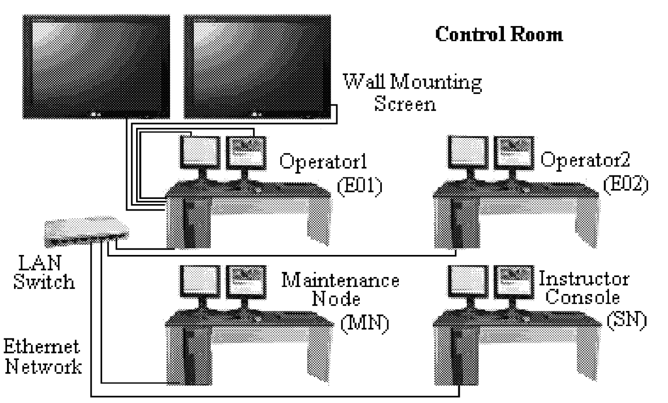


Figure 1: Hardware Architecture

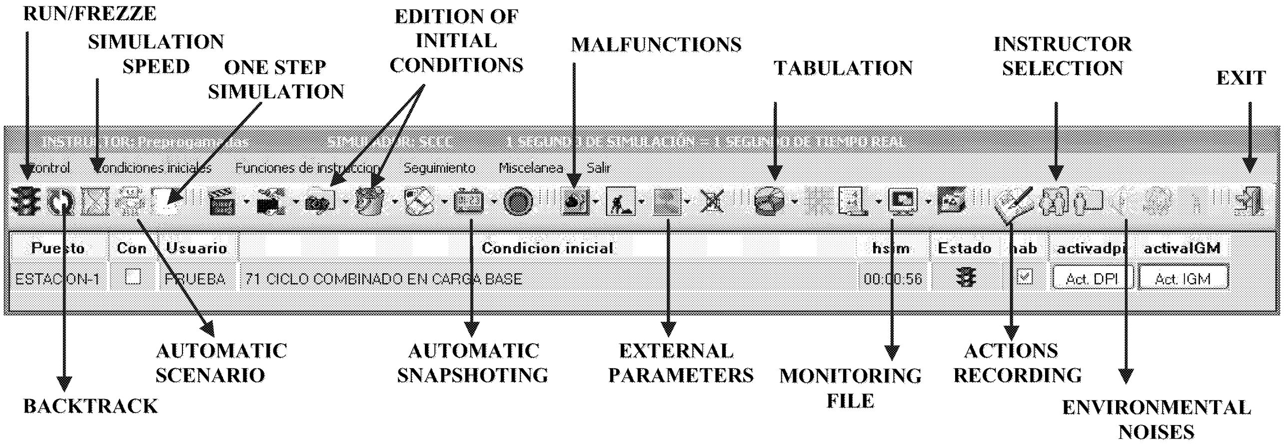


Figure 2: Instructor Console Main Menu

There is an additional PC, the Maintenance Node (MN) which is used as backup in the case the SN is out of service and to develop any modifications to the simulators. This PC has a Pentium D processor with 3.6 GHz, 2GB of RAM, 40GB HD, and runs under Windows XP. Here, the improvements are tested and validated by an instructor before to install them in the simulator.

Software

The SCC has Windows XP as an operating system, and their applications were programmed in MS C# from Visual Studio for the modules of the simulation control and interface, Fortran Intel for the mathematical models, VisSim for some control modules (being translated into C#), and Flash for the gas turbine control screens (the steam turbine controls were translated into C# and its control screens are executed in Windows form).

The simulation control and interface proprietary software of the IIE, called MAS (Jiménez-Fraustro 2005), has three main parts: the real time executive, the console module, and the operator module. Each module runs in a different PC, and all of them are communicated by a TCP/IP protocol.

The real time executive coordinates all the tasks to assure the simulator be executed in real time: the mathematical models; the interactive process diagrams; the global memory area; the Instructor Console; and the data base driver (the SCC uses about 27,000 variables).

The console module is the man machine interface that mainly executes: a module to communicate the console with the real-time executive; a module to retrieve all the static information during simulation session (malfunctions, internal parameters, local instrumentation, external parameters, and auxiliary functions); a module to store information in a data base using SQL programs as interface; the mathematical models; and the graphical interface of the instructor. In Figure 2, the main menu of the instructor console is presented indicating some of the generic functions (in the case of this simulator, no hard panels are present and the environmental noises are not simulated).

The operators' module is a replica of the control screens of the real plant to be operated. As an example of a control diagram from the operators' console, the screen for the control of the auxiliary and gland steam is presented in Figure 3.

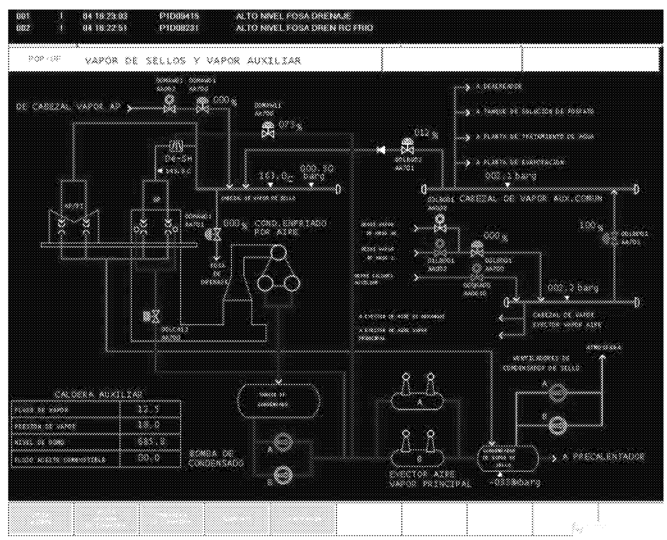


Figure 3: Control Diagram of the Modeled Steam Systems

Modeled Systems

The models were developed using a modular and lumped parameter approaches with both, storage and resistive modules (Colonna 2007) and divided in two groups: the control models (logical and analogical) and the process models. They run sequentially in one PC with two processors.

The logic control models receive and process the signals generated by actions realized by the operator in the control room via the actions on the control screens. As a response, these models generate possible changes on the process models and control screens. The models of the analogical control adjust the aperture of the valves to regulate some variables of the processes around a set point fixed by the operator (levels of tanks, pressure and temperature of certain steams, etc.). Typically there was one control model associated to one process model.

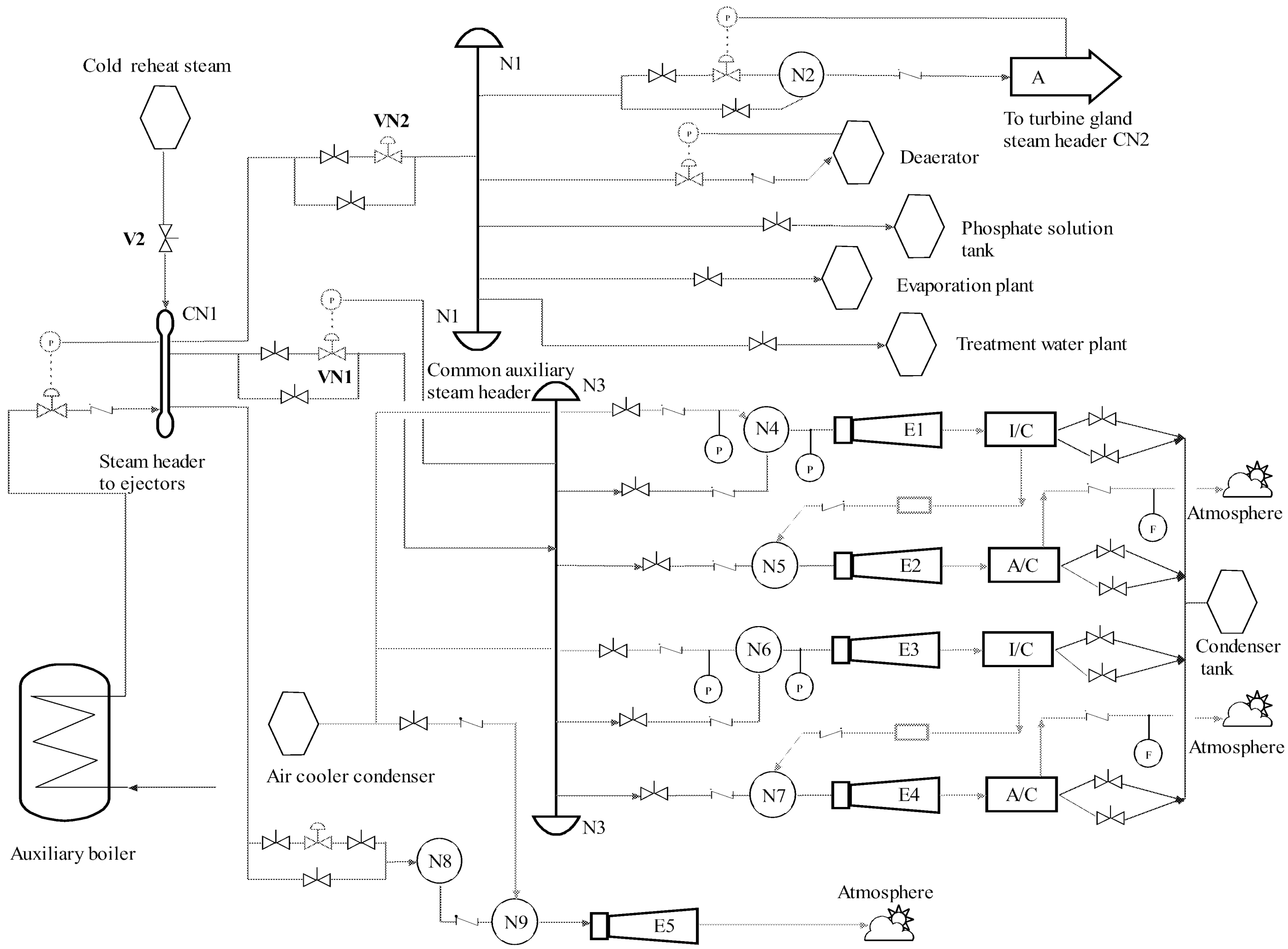


Figure 4: Auxiliary Steam Diagram, Part Services

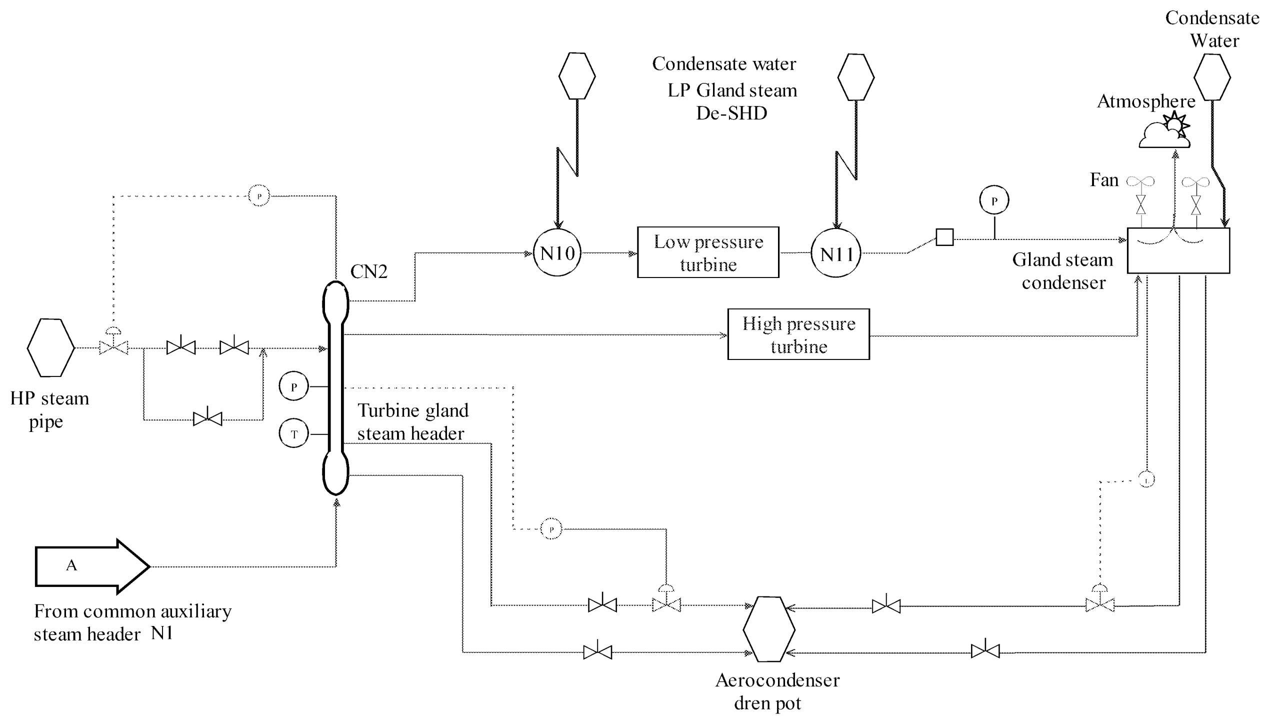


Figure 5: Simplified Diagram of the Gland Steam System

Most of the controls are part of the distributed control system (DCS) diagrams provided by the plant constructor and some of them are local controls (out of the distributed control system). A detailed description of the control development for the simulator is presented in (Romero et al. 2008).

The process models of the simulator system were developed in FORTRAN. The models are a set of algebraic and differential equations obtained from the application of basic principles (energy, momentum and mass balances).

The models were designed to work under a full range of operation conditions: from cold to 100% of load including all the possible transients that may be present during an operational session in the real plant due to the operator's actions or due to malfunctions.

The systems included in the simulator are: one full scope gas turbine unit (Roldán-Villasana et al 2008); one simplified gas turbine; condensed water, including the air-condenser; feedwater; HRSG1 & HRSG2; HP, IP, and LP water; HP, IP, and LP steam turbine; turbine metals; electric system & generator; primary and secondary regulation for the three generators; cooling water; control oil; unit efficiencies; generator cooling air; lubricating oil; chemical's dosage to the cycle water-steam; instrument and service air; HRSG1 & HRSG2 drains; steam drain; chemical analysis; potable water; services water; water of sanitary garbage; demineralized water; common and chemical drains; auxiliary steam; and turbine gland steam.

GLAND AND AUXILIARY STEAM MODELS

Systems Description

A simplified diagram of the auxiliary steam system is presented in Figure 4. The system may be divided in two functional parts, one for services and other for the main condenser vacuum (ejectors). The common part between the two functions of the system is constituted by an auxiliary boiler; a header to receive the working steam from the auxiliary boiler or from the HRSG (an automatic valve controls the pressure in the header when the steam comes from the boiler); and two distributor headers that provides steam (at a controlled pressure). From one distributor, the service part sends steam to the turbine gland steam system and other parts of the plant. The vacuum part consists of a start-up ejector and two units (one redundant) to maintain the main condenser vacuum during normal operation. Each ejector unit has two ejectors and the driven steam works in parallel from the distributor whilst the extracted air is suctioned in series from the air-condenser for the first ejector and from the inter-condenser where the first ejector discharges the steam/air mixture. A post-condenser is used to separate the water (to be sent to the main condenser) from the airs that is discharged to the atmosphere.

The turbine gland steam system seals the low and high pressure turbines in order to avoid the entrance of air or leaking of the steam, respectively. The seal steam is taken from the high pressure HRSG or from the auxiliary steam system at a controlled pressure. The gland steam header receives the steam and sends it to seal the turbine. If necessary the steam for the sealing of the low pressure turbine is deheated with water from the condensate system (to avoid material fatigue). After the sealing of the turbines, the steam is fed to the gland steam condenser and the condensed water is sent to the drain pot. A couple of extractors (fans) keep the vacuity in the gland steam condenser. A simplified diagram of this system is presented in Figure 5.

Assumptions and Simplifications

All the networks of the systems are assumed to be incompressible. The effects of changes on pressures and temperatures are concentrated in the capacitive nodes that behave partially as an ideal gas (the change of the pressure of the headers CN1 and CN2).

The difference in the height of the elements is neglected.

The Dalton's law is valid for the air/steam mixture.

The liquid and vapor of the condensers are always in thermodynamic equilibrium (saturated).

The systems are perfectly isolated and no leaks are allowed.

The pressure of the auxiliary boiler was adjusted versus the time with a prefixed the curve.

The malfunctions included in these systems are the closure or opening of the valves, misreading of the control signals, and the stuck of some valves.

Flows and Pressures Networks

The network was modeled using *flupre*, a generic model that may solve any hydraulic network (Mendoza and Roldán-Villasana 2004). This model is derived from the momentum equation applied in each stream and the continuity equation for each node. Neglecting the terms of temporal acceleration and considering the forces acting on the fluid are instantly balanced. A model may be stated integrating the equation along a stream:

$$\Delta P = -X\,\Delta\tau + \rho\, g\, H \qquad (1)$$

where P is the pressure, X the length, ρ the density, g the gravity acceleration, H the height, and τ the viscous stress tensor that may be evaluated using empirical expressions for any kind of element. For example, for pumps it may be assumed that:

$$\Delta P = K_1^{'}\, w^2 + K_2^{'}\, w\,\omega + K_3^{'}\,\omega^2 \qquad (2)$$

where K'_i are constants, w the mass flow, and ω the angular speed of the pump. For non rotating elements the equation may be stated as:

$$w^2 = K'\,\rho\, Ap^{\eta}\,\Delta P \qquad (3)$$

where the symbol Ap applies for the aperture of the valves but it may represent a resistance factor to the flow for other fittings.

The exponent of the aperture η represents the characteristic behavior of a valve.

Specifically for these steam models, the ejector where simulated using the pumps representation (eq. 2) where the speed was substituted by the driven steam flow. Eventually, the equation was modified by changing the quadratic exponent of the speed by a minor number (1.3).

Thus, applying the proper momentum equation on each element and the mass balance on each node, a system of equations is stated where the flows and pressures are the unknown variables to be solved.

So, the formulation has the form that recently has been claimed to be a novel approach to solve this kind of problem (Barry 2008). However, additionally flupre may solve any hydraulics network by detecting automatically the topology and setting and solving the appropriate set of equations. The parameters are adjusted to satisfy the design or operative conditions at 100% of load.

Mass Balances

All the steam mass balances are automatically accomplished by flupre, however, the presence of air and its propagation must be considered through the network. The air concentration is present in the HRSG during the start-up process and in the air-condenser in all conditions.

In the capacitive nodes, the concentration of the air is calculated as:

$$\frac{dc}{dt} = \frac{\sum w_i (c_i - c_o)}{m} \tag{4}$$

With the air concentration c it is possible to calculate the molar fraction of the air $\hat{c}$ and, assuming the Dalton law is valid, the partial pressure of the steam P_v may be evaluated:

$$P_v = \hat{c} P \tag{5}$$

The thermodynamic properties in the node are calculated using the partial pressure of the steam and verifying if, eventually, there is possible to have a change in the phase (condensation or evaporation).

Energy Balances

The calculations of energy balances are programmed following the flow direction and considering possible sets of equations to be solved simultaneously. In each node where energy balances are required (due to the joint of several streams), the enthalpy h is calculated by the integration of the next equation where q is the heat flow to the atmosphere and m the mass in the node:

$$\frac{dh}{dt} = \frac{\sum w_i (h_i - h_o) - q_{atm}}{m} \tag{6}$$

The heat transfer to the atmosphere of the nodes is calculated in the same way that the heat to the atmosphere of the condensers:

$$q_{atm} = U\,A\,(T_o - T_{atm}) \tag{7}$$

where the heat transfer global coefficient U and the area A depend on the physical characteristics and operative of each condenser (or node).

The gland seal steam condenser is cooled with water that flows through tubes. A countercurrent heat transfer approach was adopted to simulate this equipment but considering the condensation phenomenon. An iterative method is applied. Firstly, a heat transferred is calculated as a function of the logarithmic average temperature ΔT (an initial value of the outlet temperatures is necessary):

$$q = H\,A\,\Delta T \tag{8}$$

Here, the heat transfer coefficient H depends on the conditions and the flow rates of both streams (both calculated by flupre). Automatically the hot and cold streams are identified.

The exit temperatures of the streams are calculated as:

$$T_e = T_i + \left(\frac{q}{w\,Cp}\right) \tag{9}$$

It is necessary to verify that the temperatures do not are "crossed", i.e. the exit temperature of the cold stream does not be warmer than the inlet temperature of the hot stream and the exit temperature of the hot stream does not be colder than the inlet temperature of the cold stream, even considering the condensation of the steam (for this case the eq. 9 is calculated with enthalpies instead temperatures). If the temperatures are crossed, the heat calculated by eq. 8 is too high and has to be corrected and re-calculated the exit temperatures up to convergence.

Capacitive Nodes

The inertial effects due the compressibility of the steam on the networks are concentrated in the capacitive nodes. The capacitive nodes, for these systems, are of two types: the headers and the condensers.

The headers are considered to behave like an ideal gas and the change of the pressure is:

$$\frac{dP}{dt} = \frac{R\,T}{V}\left(\frac{dn}{dt}\right) + \frac{R\,n}{Cp\,V}\left(\frac{dh}{dt}\right) \tag{10}$$

The change in the moles n of the nodes is obtained correcting the change of the mass (being this a state variable too) with the molecular weight.

For the condensers, the derivative of the mass of liquid is obtained considering thermodynamic equilibrium between the phases (subindexes f, g and fg represent saturated liquid, saturated vapor and difference between vapor and liquid values, respectively):

$$\frac{dm_l}{dt} = \frac{1}{v_{fg}}\left(\left[m_l\left(\frac{dv_f}{dP_v}\right) + m_v\left(\frac{dv_g}{dP_v}\right)\right]\frac{dP_v}{dt}\right) + v_g\left(\frac{dm_t}{dt} - \frac{dm_a}{dt}\right) \quad (11)$$

Establishing an energy balance for the whole condenser, the derivative of the steam pressure is obtained, and here is presented in a functional way:

$$\frac{dP_v}{dt} = f\left(prop, m, M, P_v, Cp_a, v, \frac{dv}{dP_v}, \frac{dm_t}{dt}, \frac{dm_a}{dt}, \frac{dprop}{dP_v}\right) \quad (12)$$

It must be emphasized that the state variables are the liquid mass, the air mass and the vapor pressure; that the evaporation and condensation phenomena are implicit in eq. 14 as a result of pressure changes; and that in saturated conditions the derivatives of the thermodynamic properties are based on a single variable of state, in this case of the vapor pressure:

$$\frac{dprop}{dt} = \frac{\partial prop}{\partial P}\frac{dP}{dt} \quad (13)$$

IMPLEMENTATION AND RESULTS

The models were successfully implemented as a part of the simulator. The experience of the participants both, the client and the simulation department, was essential to carry out the implementation without great problems.

Perhaps the more difficult part was the validation of the models. Too many scenarios should be verified. The simulator validation was carried out proving its response against the sixteen very detailed operation procedures ("Acceptance Simulator Test Procedures") elaborated by the CNCAOI instructors who are specialized CFE personnel. These tests were the result of the experience of the instructors and the digital data obtained from the plant during diverse operation circumstances and conditions. Additionally, during the development of the models, some local tests were performed in order to detect any abnormality in the expected behavior of the models.

As an example of the validation of the model, here a test was designed to be executed with the entire simulator running. The test initiates with a full load condition. The auxiliary steam is taken from the HRSG and the gland steam from the auxiliary steam. During the first minute, no changes are induced and then the valveVN1 (that feeds the motive steam to the ejectors) is closed. At the 5th minute, the valve VN2 is closed too (that feeds steam to the auxiliary steam system from the HRSG).

It may be observed a rising of the main condenser pressure and a descent of the generator charge in Figure 6 during all the transient up to the generator trips due to a low vacuum in the main condenser (approximately at the minute 22), in this point the vacuum descent again due the fans of the main condenser. No other effects are presented in the studied systems. Also, it may be observed in Figure 6 how the vapor pressure (not the total pressure) of the gland steam condenser rises when the VN1 is closes because the augmentation of the auxiliary steam header pressure (not shown) due the diminution of its exit flowrate.

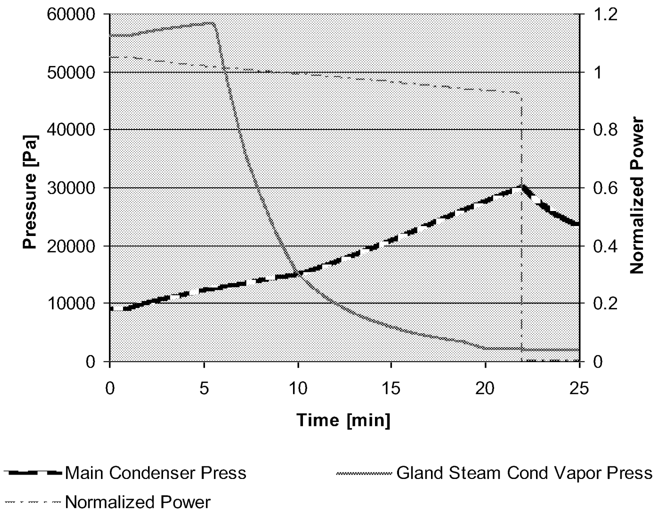


Figure 6: Pressures and Power

The pressure descents when the valve VN2 is closed and the pressure tends to the value of the saturated pressure at the ambient temperature. A similar tendency is presented by the gland steam header pressure (CN2), shown in Figure 7 (considering that it is a total pressure).

The inter-condenser pressure (in Figure 7) tends quickly to the atmospheric pressure when the valve VN2 is closed (it is a small equipment).

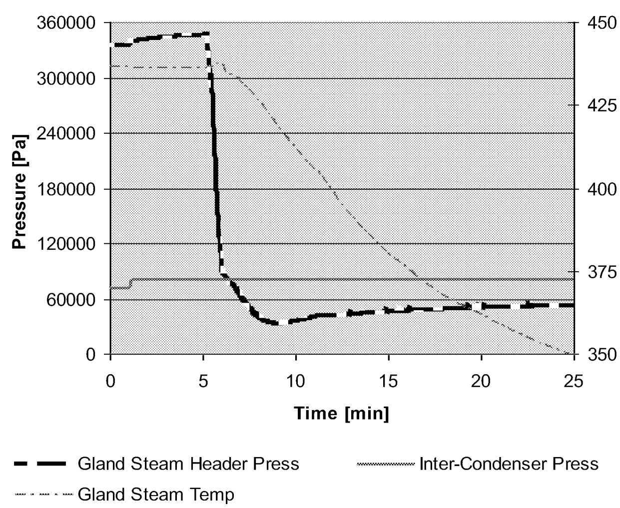


Figure 7: Pressures and Temperature

The temperature of the gland steam of the low pressure turbine descents once the cooling due the heat loosing to the ambient dominates due that the node does not receives any more flowrate (see Figure 8) because the feeding valve VN2 was closed.

In Figure 8 the aperture of the de-heat temperature control valve may be appreciated. It closes gradually due the descent of the controlled temperature (see Figure 7). The valve that controls the pressure of the gland steam header opens trying unsuccessfully of adjust the pressure.

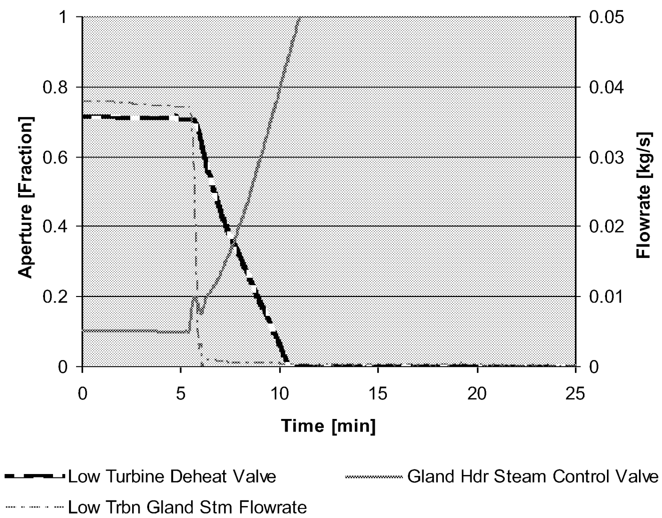


Figure 8: Apertures and Flowrate

All these results are an appropriate representation of the real systems behavior. Besides, the simulator (including these systems) was probed in all the normal operation range, from cold start conditions to full charge, including the response under malfunctions and abnormal operation procedures. In all cases the response satisfied the ISA-S77.20-1993 Fossil-Fuel Power Plant Simulators Functional Requirements norm that guarantees the simulator to be an appropriated training tool.

Future Work

The IIE is working in an automatic graphical interface to construct the models into a simulation environment.

CONCLUSIONS

The IIE has developed several simulators mainly for the training of different kind of power plants operators. The main considerations and basis for the development of two of the steam models of a full scope simulator for training purposes were presented.

The hardware and software characteristics allow the simulator to accomplish the necessities of the training requirements under the international norms.

The models were developed using basic principles and validated correlations. This allows the model to be valid in all operational ranges, including abnormal operations and malfunctions.

The local and comprehensive tests have demonstrated that the models represent the behavior of the real systems in an adequate manner.

The simulator is a replica, high-fidelity, plant specific simulator for operator training.

In the near future an automatic procedure to construct simulators will be available from the IIE.

ACKNOWLEDGMENT

Authors wish to thank all the participant IIE researches in this project and the technicians, trainers and operators of CNCAOI.

Special thanks are due to the Director of the Control Systems Division, The Head of the Simulation Department, and the Head of the SCC project.

REFERENCES

Barry J.D., "Eliminate Iteration from Flow Problems", *Chem. Eng. Progress*, March, 37-41, (2008).

Colonna P., H. van Putten, "Dynamic Modeling of Steam Power Cycles. Part I – Modeling Paradigm and Validation", *Applied Thermal Engineering*, 27, 467-480, 2007.

Jiménez-Fraustro L.A., "Desarrollo del Medio Ambiente de Simulación para Simuladores en Tiempo Real basado en sistema operativo Windows XP", *Reporte Interno, Gerencia de Simulación, IIE, México*, 2005

Mendoza, Y., Roldán-Villasana, E.J., "Methodology to Adapt the Feedwater and Condensate System, Using a Flow and Pressure Generic Model, for the Laguna Verde Nuclear Power Plant Simulator". *Proceedings of Summer Computer Simulation Conference*, The Society for Modeling and Simulation International, San Jose California, July 2004.

Molchanov, K.A, Strashnykh, Zhezherya, D.A and Manevskaya, O.A. "The full-Scale training Simulator for educating the Operative Personnel of the PGU-450 Unit at OAO Monsenergo´s TET´s-27 Cogeneration Station". *Thermal Engineering*, Volume 55, pp 886-893, (2008).

Roldán-Villasana, E.J., Mendoza, Y., Zorrilla, J., Cardoso M. and Cruz, R. , "Development of a Gas Turbine Full Scope Simulator for Operators' Training". European *Modelling Symposium, Second UKSim European Symposium on Computer Modelling and Simulation*, UK Simulation Society, Liverpool, England, 6-8 September (2008).

Romero G., Jiménez V. and Roldán-Villasana, E.J., "Graphical Environment for Modeling Control Systems in Full Scope Training Simulators", *ICMS 2008, International Conference on Modeling and Simulation*, WASET, Paris, France, July (2008).

Zabre, E., Roldán-Villasana E.J., Romero, G. , Cruz. "*Combined Cycle Power Simulation for Operator´s Training*". (Submitted for publication in World Congress on Engineering & Computer Science 2009).

BIOGRAPHY

Edgardo J. Roldán-Villasana got his BSc degree as Chemical Engineer from the Autonomous Metropolitan University (México) in 1980; his MSc degree in the Process Specialty (Chem. Eng.) from the National Autonomous University of México in 1986; and his PhD in Process Simulation from the University of Manchester, Institute of Science and Technology (UK) in 1992. Since 1980 is a researcher of the Simulation Department of the Electrical Research Institute. He is head of projects (including the development of the gas-turbine simulator) and his work is related with the areas of modeling and simulation of thermal-hydraulics processes and transport phenomena. He has published about 40 papers in international journals and congresses, including the chapter in a book.

AN ANALYSIS OF A FOSSIL UNIT FOR PRESSURE EXCURSIONS USING 1D THERMAL HYDRAULICS AND 3D CFD METHODS

Henry Sierk, PE
Dominion
5000 Dominion Boulevard
Glen Allen, VA 23060
United States of America
E-mail: Henry.Sierk@dom.com

D. Scott Lucas, Ph.D.
Phillip Wang, PE
Jessica Sierk
Microfusion Engineering Labs, Inc.
3250-E Peachtree Corners Circle
Norcross, GA 30092
United States of America
E-mail: scott.lucas@microfusionlab.com
E-mail: pwang@microfusionlab.com
E-mail: jessica.sierk@microfusionlab.com

KEYWORDS
Furnace pressure, ID fan, FD fan, damper, implosion, master fuel trip, model, THINK-4, STAR-CCM+, CFD

ABSTRACT

The testing of plant equipment for the adequate response to dynamic transients is costly and time-consuming. Simulation (staging) of the plant and its control systems and equipment offers the possibility for testing of control strategies, to reduce the probability of plant damage to near zero. This can be accomplished via time-dependent one-dimensional thermal hydraulic computations using highly accurate spatial resolution. In order to check the one-dimensional simulations, three-dimensional Computational Fluid Dynamics (CFD) has reached a point of maturity wherein the full set of Navier-Stokes equations (reality) can be used to ascertain the limits and accuracy of the 1D analysis. This new coupled procedure provides a cost-effective option in evaluating the effects of plant response to dynamic transients.

INTRODUCTION

During 1994 an under-pressure event of -52 inches of water (-131.0 mbar) at the James M. Gavin Power Station in Cheshire, Ohio came close to an implosion event for the furnace.

The excursion began with a faulty control signal that forced all the Induced Draft (ID) fan inlet guide vanes open while the plant was only operating at partial load. Forced Draft (FD) fans operating at partial load cannot supply sufficient combustion air to meet the demand if the ID fans are drawing out air at full load; so a significant air deficit was inevitable. Cross-limiting controls executed a fuel trip, since the fuel to air ratio was no longer within safety limits. Air within the furnace cooled rapidly – thus becoming denser - after the flameout, enabling the ID fans to evacuate an even greater mass of air. The limited influx of FD fan air – combined with any leakage air that had made its way in through tiny crevices – were inadequate to compensate for the immense loss of air through the ID fans (Adams 2004).

The Gavin plant was unprepared to face a fan control failure and master fuel trip – the two predominant risk factors for furnace implosion (Clelland 1977). Furnace implosion is defined as a negative pressure excursion that causes damage to furnace settings or furnace walls. Gavin's buckstays could not sustain the furnace suction, nor could the control system either prevent the pressure excursion or arrest its progression.

Multiple furnace implosion incidents earlier than, but similar to, the Gavin accident prompted the National Fire Protection Association to draft preventive regulations, NFPA 85 (Adams 2004).

In addition to requiring that furnace structural elements be built to withstand pressure excursions between (-88.0 mbar) – 35 and (+88.0 mbar) +35 inches of water (with certain exceptions), NFPA 85 6.5.2.2.2 prescribes the use of additional control elements that maneuver the system safely through transients that are impossible to manage using standard furnace pressure controls alone.

Any malfunction activating a master fuel trip (MFT) – such as low combustion airflow, high furnace pressure, loss of fuel or flame – has the potential to culminate in a furnace implosion. With the sudden absence of flame comes a precipitous fall in furnace temperature, and a proportional drop in pressure. If these developments are permitted to take their natural course, without control intervention, a dangerously low furnace pressure will result, due to the performance imbalance between FD and ID fans. Although FD fans will temporarily continue pumping in just as much air as before the trip, ID fans – if allowed to keep functioning at pre-trip levels – will remove more air than before, since the air within the furnace has become compressed by cooling. Therefore NFPA 85 6.5.2.2.2(2) requires that an MFT signal activate a feed forward action, beyond the auto/manual station, that immediately reduces ID fan demand (usually by a third of its pre-trip level).

Industry experience has shown that when furnace implosion occurs, it is usually due to a combination of factors - multiple failures in addition to the effects of a Master Fuel Trip (MFT).

Due to limitations in the resolution capabilities of transmitters, furnace pressure transmitter ranges cannot be set to include the lowest or highest possible values (e.g. +/-35" WC [+/- 87 mbar] or more). Normal furnace pressure ranges are typically much narrower, such as +/- 10-15" WC (26 to 40 mbar). Thus once a monitored value falls outside the transmitter span, as would happen during a severe pressure excursion, the controller no longer receives an accurate current value from the transmitter. So NFPA 85 6.5.2.2.2(3) dictates the addition of fan override action, beyond the auto/manual station, which adds more proportional control so that furnace pressure can be monitored and controlled effectively even when it falls outside the range of the transmitter.

NFPA 85 6.5.3.3.5 also states, "Following shutdown of the last fan due to any cause, the opening of fan dampers shall be delayed or controlled to prevent positive or negative furnace pressure transients in excess of design limits during fan coastdown."

Ordinarily, if there are two or more FD and ID fans, fans are controlled in pairs, such that if one FD fan trips, its corresponding ID fan will automatically trip in response, and all vanes or dampers on both stopped fans will automatically be closed. The loss of all ID or all FD fans will initiate a master fuel trip, but will also open all fan vanes and dampers, after a time delay to allow for the fans to coast down in speed.

ID fan dampers take several seconds to close. The motor of an ID fan is shut off immediately upon receiving a trip signal, but inertia dictates that the fan blades will coast down to a stop, rather than coming to an abrupt halt. Therefore during a pressure excursion, travel time slows the actuation of protective control signals.

Risk increases if there is only a single ID fan and multiple FD fans, or if the coastdown period of an ID fan exceeds that of its associated FD fan. This is the situation at Dominion's Chesterfield Power Station, due to environmental upgrades that are currently in progress.

Dominion plans to replace two ID fans with a single, higher-performance ID fan in its Chesterfield Unit 4 coal-fired power station, which has two FD fans. The Unit 4 ID fan will discharge into a common header shared with Units 3 and 5. Furnace pressure will be controlled using the ID fan's variable frequency drive (VFD) as the final control element. The ID fan inlet damper will be manipulated by the controller as a backup to the VFD.

The rotary moment of inertia of the new Unit 4 ID fan is 161,637 lb ft^2, more than double that of the prior two ID fans together (60,533 lb ft^2 x 2), so its coastdown period will be longer. This installation presents significant new hazards of negative pressure excursion. For instance, if the FD fans were to trip, then the ID fan would trip in response, and the lengthy coastdown period of the ID fan could lead to dangerously low furnace pressure.

Consequently, Dominion needed to perform extensive pre-installation testing of this plant configuration to determine an effective anti-implosion control strategy. They were faced with two options for testing: field implementation testing or simulation. Field implementation testing was determined to be too risky and expensive. Comparatively, simulation was a safe and comprehensive solution. Simulation provided the option of testing dozens of potential scenarios in order to generate what would be the furnace pressure response to each scenario. Dominion decided that their control strategies would be developed on the basis of the simulation.

SOLUTION STRATEGY

One-dimensional Computational Fluid Dynamic (CFD) code such as THINK-4 provides the advantage of being able to simulate various transient scenarios in a relatively short time. The disadvantage of one-dimensional code is that it only provides the average pressure and flue gas velocity of each lumped (node) section of the furnace.

The advantage of three-dimensional CFD code such as Star CCM+ is that it determines the pressure drops and flue gas flow distributions in detail, throughout the furnace and the ductworks, while the time involved in running each simulation scenario is significantly longer than for the 1D simulation.

THINK-4 is a four equation non-equilibrium thermal hydraulics code which allows for the modeling of primary fossil and nuclear components along with Balance of Plant (BoP) models. It also models non-condensible gases, such as air, in or out of liquid solution.

Star-CCM+ is a Navier-Stokes based commercial CFD code. It uses the Finite Volume Method (FVM) to integrate the equations in one, two, or three dimensions and determines the pressure drops and flow distributions more accurately using experimentally tested turbulence models.

A strategy that utilizes the advantages of both the 1D and 3D CFD codes was adopted. In this approach, the furnace simulation models were developed using both 1D (THINK-4) and 3D (Star CCM+) CFD Codes. The majority of the scenarios simulations were performed using the 1D CFD code after selected initial results of the 1D code were validated with the 3D CFD code.

THINK-4 MODEL

The system, shown schematically in Figure 1, was partitioned into four sections in THINK-4. The conservation laws governing mass, momentum, and energy conservation along with the equations of state describing material properties were applied. These four modeled sections in the THINK-4 model are:

1. Forced Draft Fan, Inlet Ducts, and Windbox
2. Furnace and Combustion
3. Scrubbers and Precipitators
4. Stack and Induced Draft Fan and Damper

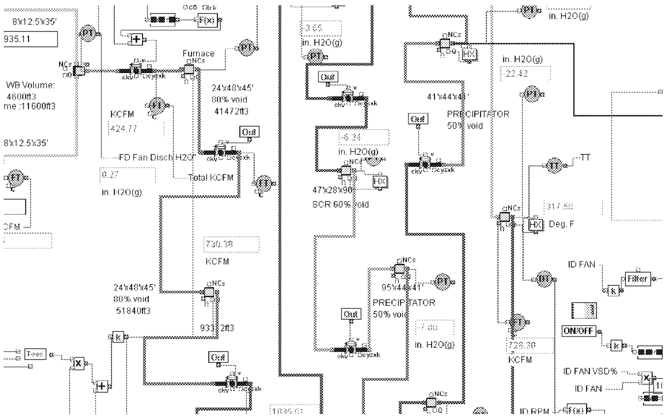


Figure 1: THINK-4 Nodal Diagram (Partial)

The major components modeled were:

1. Forced draft fan (including inertia) and vanes
2. Forced draft fan outlet duct
3. Air heater and tempering coils
4. Windbox
5. Back pass
6. Gas recirculation fan and ducts
7. ID fan inlets and crossover duct
8. ID fans (including inertia) and stack

All parameters used in the model were obtained from physical data and calculated from acceptance test data.

The starting point was the pressure drop information in Chesterfield Unit 4 PI data pressure drop across SCR and at the ID fan inlet suction. This procedure was performed from the inlet FD fans to the stack. The pressure reference points are:

- Windbox:	+0.5" WC
- Furnace:	-0.5" WC
- Economizer Outlet:	-3.5" WC
- SCR Outlet:	-5.5" WC
- ID Suction:	-18" WC
- ID Discharge:	+7" WC

Using this model, various fan-related transient conditions that could generate pressure excursions, both under cold or hot furnace conditions, were simulated.

THINK-4 Simulation: Cold Furnace Conditions

The first scenario investigated with THINK-4 was all fans tripped, with the isolation damper closing at different intervals.

The simulation was performed once with the ID fan isolation damper remaining open, and four times by adjusting the travel time required for the damper to close. The furnace pressure dropped to its lowest point, -15.7" WC, when the isolation damper stayed open. As the damper drive speed increased, the magnitude of negative pressure decreased. The negative pressure excursion abated slightly when the damper closed within 30 seconds (-15.5" WC) or 20 seconds (-15.2" WC).

Once the damper closing time was reduced to 15 seconds, the furnace pressure dropped to –7.2" WC. A smaller pressure excursion, –4.2" WC, was achieved when the damper closed in 10 seconds.

The second scenario simulated was the start of the ID fan, followed by the start of an FD fan.

Initially, all the FD fans will be off, and then after an ID fan has been started, an FD fan may be started, per NFPA 85 guidelines (6.5.3.3.1 (1)). Because any abrupt addition or subtraction of airflow can cause a pressure excursion, NFPA 85 states, "Where starting and stopping fans, the methods employed and the manipulation of the associated control elements shall *minimize* furnace pressure and airflow excursions" (6.5.3.3.2).

Figure 2 shows typical furnace pressure behavior of this operational sequence (Wang and Lucas 2009).

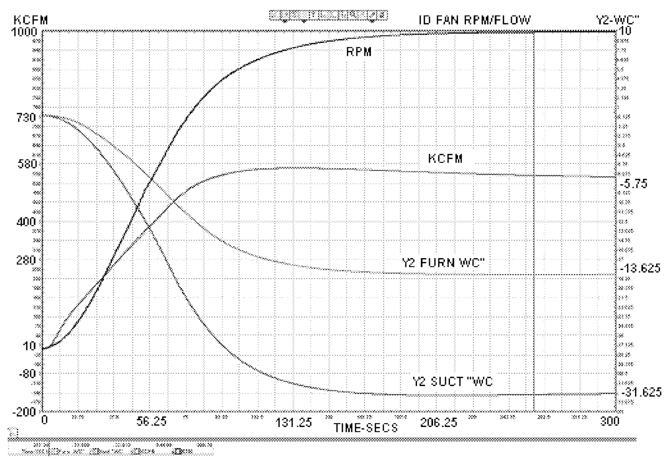


Figure 2: Cold ID Fan Start, FD Fans Off, 0" WC Discharge Pressure

Because the desired limiting ranges are +/-8" WC, new control strategies will be developed to prevent the predicted pressure excursions on startup. The effectiveness of the new strategies will be tested using the model prior to plant installation.

The third scenario analyzed was the FD fans tripping in sequence.

In this scenario, after both FD fans have tripped, the ID fan will automatically trip in response. The simulation demonstrates that furnace pressure will settle at –6.6" WC after the first FD fan trips, then level out at –10.6" WC after the second FD fan trips (Wang and Lucas 2009).

Based on this analysis, Dominion will add a feed-forward runback to the ID fan, forcing it to drop its speed significantly after the first FD fan trips. If the speed decrease fails to keep furnace pressure within safe limits, the ID fan will receive a signal to partially close its inlet dampers.

THINK-4 Simulation: Hot Furnace Conditions

Simulation 1: One FD Fan Tripped

In a plant with two or more FD and ID fans, if one FD fan trips, its corresponding ID fan will trip in response. Since Chesterfield Unit 4 has two FD fans and only one ID fan, the ID fan will only trip after both FD fans have tripped. A hazardous situation can result if only one FD fan trips, since the ID fan will still be drawing out air as though both FD fans were in service.

The ID fan discharge header was set at 11"WC to reproduce a condition in which the ID Fan discharge header pressure was above ambient (to simulate the pressure in the common ID fan discharge header shared with two other units). The simulation demonstrated that the furnace pressure will drop to –7.5" WC after a single FD fan trips (Wang and Lucas 2009).

Therefore Dominion has designed a control strategy that slows the ID fan proportionally, as stated under Cold Furnace Conditions.

Simulation 2: All Fans Tripped

The trip of all fans activates the Master Fuel Trip (MFT) relay. An MFT is followed by a furnace purge, due to an interlock which prevents re-firing the boiler until it has been purged of combustibles. Ordinarily the FD and ID fans perform the purge.

Some plants employ natural draft to purge the boiler after all fans have tripped, but this is not feasible at Chesterfield Unit 4, where the flue gas discharges into a common header shared by two other units (rather than directly to a stack). Instead, all FD fan dampers open, and then a vent fan (that discharges directly into the atmosphere) automatically starts, creating a pressure differential that mimics natural draft.

Three potential cases were simulated.

(1) In the first simulated case, the vent fan is out of service, and the unit is isolated from the common discharge header. Without an operating vent fan, there is no way to perform a furnace purge. The simulation shows that furnace pressure will spike up to 7.3" WC before going through a first order decay back to zero.

(2) In the second simulated case, the vent fan is running, and the unit is still isolated from the common discharge header. Since the vent fan discharges to the atmosphere rather than to the common header, the purge can occur. The simulation indicates that the furnace pressure will spike up to 6.1" WC before undergoing a first order decay back to zero.

(3) In the third simulated case, the vent fan is running and the unit is no longer isolated from the common discharge header (at 11" WC) but the furnace purge does not occur. Flue gas from units 3 and 5 will flow backwards through the ID fan into the furnace, and out through the FD fan inlets. After this initial drop, furnace pressure will settle at 4.8" WC (Wang and Lucas 2009).

Dominion has done its utmost to ensure that its vent fan is reliable, even to the point of bypassing motor overload protection when the vent fan is started in this mode. However, since this is the only protective measure taken, it appears to be a weak link in the strategy and more analysis may be needed.

Simulation 3: FD Fan(s) Tripped After Master Fuel Trip

FD fan output should not change following a master fuel trip, because the inflow of air is necessary to conduct the furnace purge. As mentioned previously, the trip of a single FD fan will not cause the ID fan to trip. A dangerously low furnace pressure may result if an FD fan trips and the ID fan continues suction, drawing out air at its previous rate.

Four potential cases of a single FD fan trip are simulated.

(1) In the first simulated case, the discharge header pressure is defined as 1.3" WC, to replicate a condition in which Units 3 and 5 are tripped. The simulation demonstrates that furnace pressure will drop to –16.2" WC after the FD fan trip. (See Figure 3.)

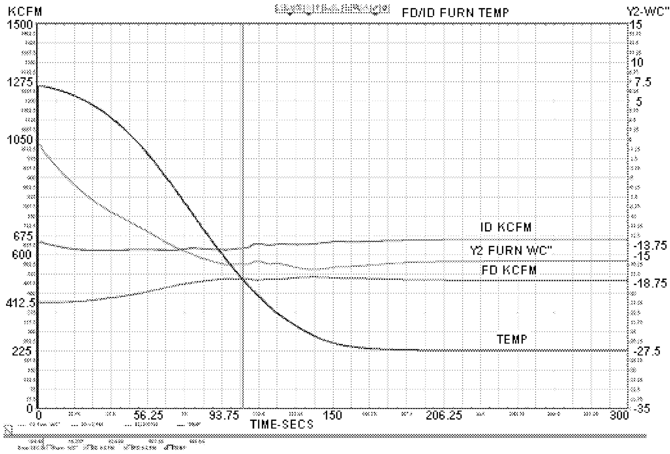


Figure 3: Hot MFT, 1 FD Fan Trip,
Units 3 & 5 Tripped, 1.3” WC Discharge Pressure

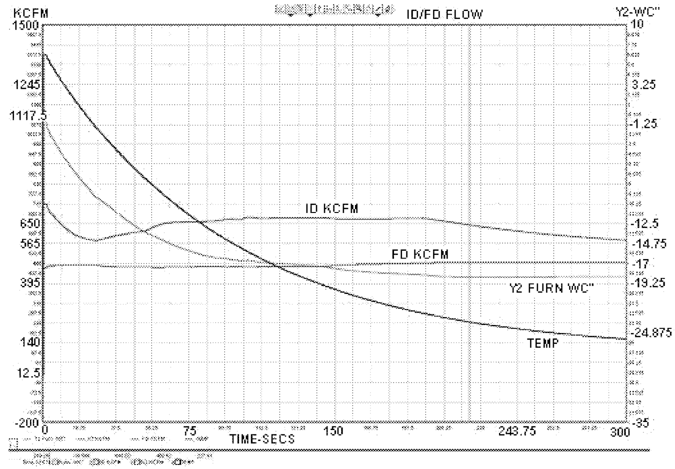


Figure 4: Hot MFT, Cooling, 2 FD Fans Trip,
0” WC Discharge Pressure

(2) In the second case, the discharge header pressure is set at 4” WC, to emulate a condition in which Unit 5 is tripped. The simulation indicates that furnace pressure will drop to –15.0” WC after the FD fan trip.

(3) In the third simulated case, the discharge pressure is specified as 0” WC, to represent a condition in which Units 3 and 5 are out of service. The simulation shows that furnace pressure will drop to –14.2” WC after the FD fan trip.

(4) In the fourth simulated case, the discharge pressure is identified as 11” WC, to reflect a condition in which units 3 and 5 are in service. The furnace is still cooling after the master fuel trip. The simulation shows that the furnace pressure will drop to –10.9” WC after the FD fan trip (Wang and Lucas 2009).

Two potential cases of both FD fan trips are simulated.

The ID fan will trip in reaction to the trip of both FD fans. Once the FD fan blades have come to a stop, their inlet dampers will open, and the vent fan will automatically start. The furnace is still cooling.

(1) In the first simulated case, the discharge header pressure is specified as 0” WC, to imitate a condition in which units 3 and 5 are out of service. The lowest furnace pressure of all the simulated events occurs under these circumstances. The furnace pressure drops to –18.6” WC after the FD fans trip, shown in Figure 4.

(2) In the second simulated case, the discharge header pressure is set at 11” WC, to represent a condition in which units 3 and 5 are in service. The simulation indicates that furnace pressure will drop to –14.7” WC after the FD fans trip (Wang and Lucas 2009).

Due to the severe pressure excursions predicted during this transient, Dominion will evaluate the design of its existing furnace, to ensure that it is able to withstand such excursions. If not, mitigating controls will need to be instituted. Regeneration may need to be added to the variable frequency drive (VFD), or the fast-acting isolation dampers may need to be closed while the ID fan is coasting down.

Simulation 4: All Fans Running After Master Fuel Trip

After a master fuel trip, the fans will conduct a purge as the furnace cools. Two potential cases are simulated. The cases, with units 3 and 5 both offline and online, generated pressures within a controllable range, so they did not require any special action (Wang and Lucas 2009).

STAR-CCM+ MODEL

The Star-CCM+ model of the Chesterfield Unit is based on a 3D CFD full Navier-Stokes simulation. The model is meshed using 500K polyhedral cells, shown in Figure 5.

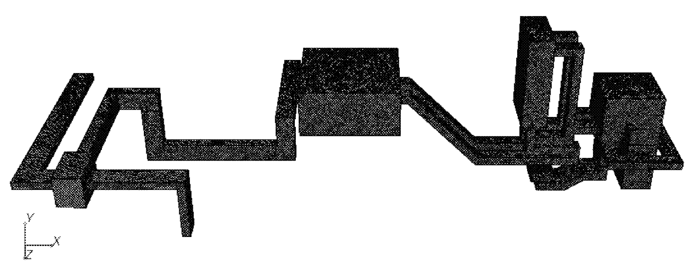


Figure 5: 3D Model Mesh % Geometry

The model was used to examine the limit or bounding analysis of transients for THINK-4, to ensure that the calculations were best-estimate but also conservative. The Star-CCM+ analysis typically showed less pressure excursions, either positive or negative, for the transients. This is due to using a k-ε turbulence model that allows the dissipation of energy more effectively. Since the thermal hydraulic equations are a subset of the Navier-Stokes equations, the Star-CCM+ simulation allows a full three dimensional distribution versus that of 1D in THINK-4. The model was also useful in examining pressure drops for realism. Figure 6 illustrates the pressure distribution for cold conditions. Figure 7 shows the velocity distribution.

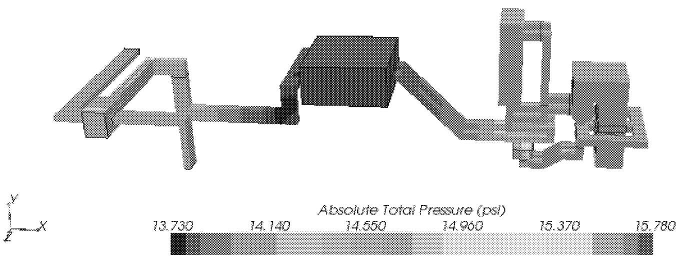


Figure 6: Cold Conditions Pressure Distribution

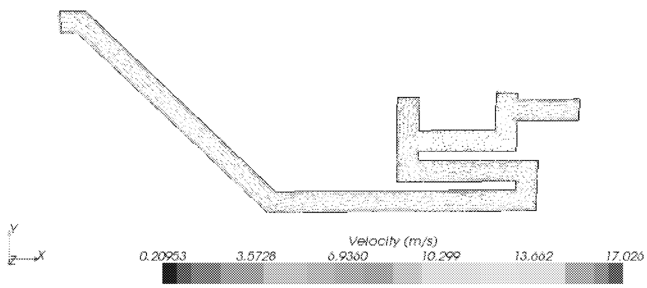


Figure 7: Cold Velocity Distributions

Combining this information with plant data allowed a more realistic steady state pressure distribution as an initial condition for running the THINK-4 dynamic simulations. Many of the cases analyzed by THINK-4 were simulated with Star-CCM+ for confirmatory analysis using the code in model on an eight CPU portable Linux box. This advance in CFD allows a more rigorous confirmatory suite of analyses rather than relying on one-dimensional simulations of the past.

CONCLUSION

The THINK-4 and Star-CCM+ models can accurately replicate the dynamics of the furnace air/gas system at Chesterfield Unit 4, yielding reliable predictions about furnace pressure response to various transients. On the basis of these predictions, Dominion has decided to:

1. purchase ID fan isolation dampers that close in 4-6 seconds
2. develop control strategies to prevent severe pressure excursion during fan startup
3. add a feed-forward runback to reduce the speed of the ID fan, along with a backup signal to close the inlet damper, after a single FD fan trips
4. evaluate the existing furnace's ability to sustain the lowest predicted furnace pressure, and institute mitigating controls if necessary

Dominion will then use the models to test each of these new control configurations, once they are designed, to assess their effectiveness. Simulation performs double-duty by both predicting plant behavior and testing new schemes developed to control potentially damaging behavior. It would be difficult to find a more economical, yet profitable solution for control logic development.

REFERENCES

Adams, R.G. 2004. "Do Fans Cause Furnace Implosions?" In *International Conference on Fans* (London, UK, Nov.9-10). John Wiley & Sons, Chichester, England, 213.

Clelland, P.J. 1977. "Recent Furnace Draft Control Considerations". In 1977 IEEE Conference on Decision and Control, including the 16th Symposium on Adaptive Processes (New Orleans, LA, Dec.7-9). IEEE, New York, NY, 1101-1102.

National Fire Protection Association. 2007. *NFPA 85: Boiler and Combustion Systems Hazards Code*. NFPA, Quincy, MA.

Wang, P. and D.S. Lucas. 2009. "The Analysis of Chesterfield Unit 4 ID Fan Dynamics." Technical Report Dom-37-01. Dominion Virginia Power, Glen Allen, VA. (Feb)

AUTHOR LISTING

AUTHOR LISTING