



# Canadian Artificial Intelligence Intelligence Artificielle au Canada

Spring/Summer 1996

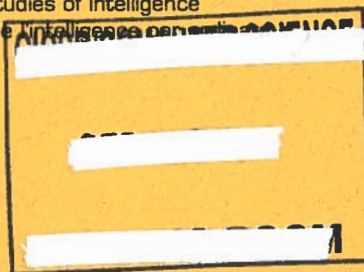
No. 39

printemps/été 1996

An official publication of CSCSI, the Canadian Society for Computational Studies of Intelligence  
Une publication officielle de la SCEIO, la Société canadienne pour l'étude de l'intelligence

**Special Issue**  
**Equipment Diagnostics**

**Edition speciale**  
**La Diagnose d'équipement**



Cover illustration by Paul Amireault



# Canadian Artificial Intelligence

# Intelligence Artificielle au Canada

Spring/Summer 1996

No. 39

printemps/été 1996

## Canada's National AI magazine

### Editors · Rédacteurs en Chef

Peter Turney and Sue Abu-Hakima

### Editor Emeritus · Rédacteur Emeritus

Roy Masrani

### Managing Editor · Chef de Pupitre

Arlene Merling

### Production Team · Equipe de production

Arlene Merling, Greg Klymchuk, Carol Tuma

### Academia · Académiques

Vacant

### Canadian AI Success Stories

Récits de Succès d'IA au Canada

Peter Turney

### PRECARN Update · Nouvelles de PRECARN

Jean-Claude Gavrel

### Book Reviews · Critiques de livres

Graeme Hirst

### Translation · Traduction

Michel Addison, Alain Desilets, Benoit Farley

### Advertising · Publicité

Arlene Merling

Canadian Artificial Intelligence is published three times a year by the Canadian Society for Computational Studies of Intelligence (CSCSI). *Intelligence Artificielle au Canada* est publiée trimestriellement par la Société canadienne pour l'étude de l'intelligence par ordinateur (SCEIO). Canadian Publications Mail Product Sales Agreement No. 507032.

ISSN 0823-9339

Copyright © 1996, Canadian Society for Computational Studies of Intelligence. All rights reserved: *Canadian Artificial Intelligence* may not be reproduced in any form without the written permission of the editors. Printed in Canada by Riley's Reproductions & Printing Ltd. *Canadian Artificial Intelligence* is published with the assistance of the Alberta Research Council. The opinions expressed herein are those of their respective authors and are not necessarily those of their employers, CSCSI, *Canadian Artificial Intelligence*, the editors, CIPS, the National Research Council or the Alberta Research Council.

Copyright © 1996, Société canadienne pour l'étude de l'intelligence par ordinateur. Tout droit réservé: *Intelligence artificielle au Canada* ne doit être reproduite par quelque moyen que ce soit sans le consentement écrit des éditeurs. Imprimé au Canada par Riley's Reproductions & Printing Ltd. *Intelligence artificielle au Canada* est publiée avec l'aide du Conseil de Recherche de l'Alberta. Les opinions exprimées dans ce magazine sont celles de leurs auteurs respectifs et non pas nécessairement celles de leurs employeurs, de la SCEIO, de l'*Intelligence artificielle au Canada*, des éditeurs, de l'Association canadienne informatique, de Conseil National de Recherches du Canada, ou du Conseil de Recherche de l'Alberta.

## CSCSI Executive

### President · Président:

Stan Matwin, Computer Science Dept., University of  
Ottawa, Ottawa, ON K1N 6N5; stan@csi.uottawa.ca

### Past-President · Président Précédent:

Janice Glasgow, Dept. of Computing & Information  
Science, Queen's U., Kingston, ON K7L 3N6;  
janice@qucis.queensu.ca

### Vice-President · Vice-Président:

Allan Jepson, Department of Computer Science,  
University of Toronto, Toronto ON M5S 1A4;  
jepson@vis.utoronto.ca

### Secretary · Secrétaire:

Fred Popowich, School of Computing Science, Simon  
Fraser University, Burnaby BC V5A 1S6;  
popowich@cs.sfu.ca

### Treasurer · Trésorier:

Peter van Beek, Dept. of Computing Science, U of A,  
Edmonton, AB T6G 2H1; vanbeek@cs.ualberta.ca

### Editors · Rédacteurs en Chef

Peter Turney, Institute for Information Technology, NRC,  
Ottawa, ON K1A 0R6; peter@ai.iit.nrc.ca

Sue Abu-Hakima, Institute for Information Technology,  
NRC, Ottawa, ON K1A 0R6; sue@ai.iit.nrc.ca

## Contents

### Communications

### Feature Articles

Diagnosis Research in Canada  
Suhayya Abu-Hakima

1 Diagnostics R&D in the  
Integrated Reasoning Group  
at the National Research Council's  
Institute for Information Technology  
Mike Halasz

PUMP - An Integrated Maintenance  
Scheduling and Costing System  
Steve Reilly

Hybrid Intelligent Systems  
and Diagnostic Applications  
Sumit Virmani and Larry Medsker

A Fault Diagnosis Knowledge-Based System  
for Manufacturing Equipment  
in a CIM Environment  
Aldo Dagnino and Brian Schack

M3i Systems Inc.  
Daniel Puche

Generic Integrated Intelligent System  
Architecture for Fault Diagnosis  
and Maintenance  
Ming Rao and Qun Wang

### PRECARN Update

### Book Reviews

*Canadian Artificial Intelligence* welcomes submissions on any matter related to artificial intelligence. Please send your contribution, electronic preferred, with an abstract, and a short bio to:

Dr. Peter Turney  
Co-Editor, *Canadian Artificial Intelligence*  
Institute for Information Technology  
National Research Council Canada  
M-50 Montreal Road  
Ottawa, Ontario, Canada  
K1A 0R6 or — peter@ai.iit.nrc.ca  
Telephone (613) 993-8564 Fax (613) 952-7151

Book reviews and candidate books to review should be sent to:

Graeme Hirst  
Canadian Artificial Intelligence  
Department of Computer Science  
University of Toronto  
Toronto, Ontario, Canada  
M5S 1A4 or — gh@ai.toronto.edu

## Contenu

### Communications

### Gros Titres

4 Recherche en diagnostic au Canada  
Suhayya Abu-Hakima

8 La R&D en diagnostic dans le Groupe de  
raisonnement intégré l'Institut de technologie  
de l'information du Conseil national de  
recherches du Canada  
Mike Halasz

13 PUMP - un système intégré de planification et  
d'évaluation des coûts de l'entretien  
Steve Reilly

20 Systèmes intelligents hybrides et  
applications au diagnostic  
Sumit Virmani and Larry Medsker

27 Un système à base de connaissances pour  
le diagnostic des pannes d'équipement  
manufacturier dans un environnement FIO  
Aldo Dagnino and Brian Schack

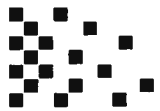
33 Systèmes M3i inc.  
Daniel Puche

35 Architecture de système intelligent intégré  
générique pour le diagnostic de panne  
et pour l'entretien  
Ming Rao and Qun Wang

### Nouvelles de PRECARN



Recycled / Recyclable




## Special Issue: Equipment Diagnostics

*Mike Halasz*

Diagnostics continues to be an active area of Artificial Intelligence. For this reason, we present a special issue on the topic. Problem solving in this area typically requires a high degree of familiarity with the diagnostic technique and respective domain. Techniques range from data analysis algorithms to model-based reasoning and domains vary from aircraft to manufacturing processes. The most robust solutions require an integration of algorithms and techniques (often referred to as hybrid solutions). These techniques are not restricted to those commonly found in AI. For example, signal analysis, not considered a branch of AI, is key to some hybrid diagnostic systems. Furthermore, in order to be truly effective, a diagnostic system must be integrated with the databases and various systems currently in use within an organization. Such work is not considered pure AI work but is essential in ensuring the success of the diagnostic system.

This issue takes a look at some of the projects going on in Canada and includes an article which reviews some

diagnostic applications which take the hybrid approach. This is by no means an exhaustive review of the state of the art but rather focuses on practical problems and the role AI plays in developing such systems. Also included in this issue is an overview of diagnostic research going on at various universities and institutes in Canada.

As is the tradition of Canadian AI magazine, the style of the articles is unrestricted. Most articles come from industry or applied research institutes. They include diagnosis applications in manufacturing, aerospace, mining, power plants, coastal vessels, forestry and wood products, as well as the pulp and paper Industry. Many thanks go to the contributors. Hopefully the readership will find this issue interesting and can result in some meaningful interactions between the variety of players. 

---

## Communication from the President

*Stan Matwin*


My term as President of the CSCSI, as well as the term of the current Executive, is coming to a close. The Executive has worked hard to achieve the goals that we set for ourselves at the beginning of our mandate. These goals focused on the problems of membership, and on the production and cost of the magazine. We have consolidated the membership, and it seems that it has now stabilized at just over 200 members. We have redefined our fee structure to attract more student members, and have made an effort to improve and streamline communication with the membership. The costs of the magazine have been decreased considerably with only a slight change in quality, resulting in a less glossy, but recyclable publication, in line with many other groups and societies.

I would like to thank the Executive for working as a team, providing their time, skills, and energy to make this a successful experience for all of us. Fred Popowich as the Secretary of the Executive contributed his energy and enthusiasm. Peter van Beek as the Treasurer provided timely and accurate financial reports, and made us aware of the financial constraints and circumstances between reports. Peter Turney and Sue Abu-Hakima ran the magazine through the difficult transition period, and contributed many interesting ideas on modernizing its contents and distribution.

Finally, I would like to thank Arlene Merling for her work as the Managing Editor of *Canadian Artificial Intelligence* magazine, and her efforts in converting the magazine to the new format.

My last task is to nominate, on behalf of the retiring executive, the new Executive. We propose the following team:

Fred Popowich, Simon Fraser University, President  
Renee Elio, University of Alberta, Vice-President  
Guy Mineau, Universite Laval, Secretary  
Howard Hamilton, University of Regina, Treasurer  
Sue Abu-Hakima, NRC, Magazine Editor (Peter Turney has agreed to continue until issue #40).

Members are encouraged to provide alternative nominations. Recommendees must be members of CSCSI/SCEIO in good standing, and must be willing to serve as officers for the two-year period. Nominations should be sent to me to be received by September 15, 1996. The election for officers will be held by mail. Ballots will be sent out September 30, 1996 and must be returned to me by November 6, 1996. If no further nominations are received, the team proposed above will become the new executive by acclamation. 



# Canadian Society for Computational Studies of Intelligence

## *Treasurer's annual report for the 1995 fiscal year*

CSCSI had two major financial activities in fiscal year 1995 — the production of *Canadian Artificial Intelligence* magazine and partial sponsorship of the 14th International Joint Conference on Artificial Intelligence (IJCAI), held in Montreal. Not considering the windfall from our share of the IJCAI profits, the society again ran a small deficit in 1995, even though we only published two issues of the magazine. Steps have been taken to lower the costs of producing the magazine, starting with the first issue in 1996, to bring expenses in line with expected future income. In 1996, CSCSI will again publish three issues of the magazine.

### **Income**

Share of IJCAI profits	\$ 11,465.94
Membership dues	5,493.87
Magazine advertising <sup>1</sup>	
CAI #36, Winter '95	800.00
CAI #37, Spring/Summer '95	800.00
GST collected	437.47
GST refund from government	275.75
Interest on bank account and GIC	1,223.37

**Total Income** \$ 20,496.40

### **Expenses**

Administration fee for collecting dues <sup>2</sup>	\$ 1,274.00
Magazine production	
CAI #36, Winter '95	4,354.54
CAI #37, Spring/Summer '95	4,646.47
GST paid out	536.95
GST paid to government	176.27

**Total Expenses** \$ 10,988.23

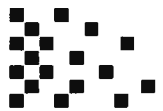
**Year Total** \$ 9,508.17

**Bank balance as of 31/12/95 (incl. GIC)** \$ 39,192.97

### **Notes**

1. I have decided to present the magazine income and expenses according to when the issue was published, rather than in the year that the money is actually received or spent.
2. The Canadian Information Processing Society (CIPS) handles our membership renewals and charges us a fee of \$4.00 for renewals who are also members of CIPS and \$10.00 for renewals who are not members of CIPS. In 1995 there were 170 renewals (71 were members of CIPS and 99 were not members of CIPS). In comparison, in 1994 there were 246 renewals (95 were members of CIPS and 151 were not members of CIPS).

*Peter van Beek*



## **Diagnosis Research in Canada**

*Suhayya Abu-Hakima*

### **Introduction**

Cet article regroupe des résumés fournis par des chercheurs dans le domaine du diagnostic au Canada, qui illustrent ce qui se fait dans ce domaine sur les plans théorique, expérimental et appliqué. Les travaux varient d'une théorie de la résolution de problème pour le diagnostic, à l'université de Toronto, au développement d'un système de diagnostic neuromusculaire en utilisant des réseaux de Bayes, à l'université de la Colombie britannique.

On a demandé aux chercheurs de fournir de brefs résumés (certains le sont plus que d'autres) et des renseignements sur les personnes à contacter pour en savoir plus long. Si vous êtes intéressés par un champ d'activité en particulier, je vous encourage fortement à entrer en contact directement avec les chercheurs pour obtenir des références ou des détails supplémentaires.

### **Introduction**

This article is a collection of summaries provided by researchers in the diagnosis field in Canada. It illustrates the scope of the theoretical, experimental and applied work. The work ranges from developing a theory for problem solving for diagnosis at the University of Toronto to developing neuromuscular diagnosis using bayesian networks at the University of British Columbia.

The researchers were asked to provide brief summaries (some are more brief than others) as well as contact information for follow-up. If you are interested in a specific area of work, I urge you to contact the researchers directly for references or further details.

### **Experience Aided Diagnosis**

*Janice Glasgow*

Department of Computing and Information Science  
Queen's University, Kingston, Ontario

Michel Feret and Janice Glasgow have designed and implemented a novel approach to model-based diagnosis that incorporates case-based reasoning to evaluate and characterize errors in complex mechanical devices. This work was sponsored by the Canadian Space Agency and carried out in collaboration with Spectrum Engineering.

Contact: Janice Glasgow

E-mail: [janice@qucis.queensu.ca](mailto:janice@qucis.queensu.ca)

### **Theoretical, Methodological and Experimental Diagnosis**

*Bechir el. Ayeb*

Faculty Sciences/DMI  
Université Sherbrooke, Quebec City

RiFa laboratory at the University of Sherbrooke focuses on different aspects of diagnosis research, including theoretical, methodological, and experimental issues. Professors Ayeb and Wang are the co-directors of the lab. Seven researchers are involved in the following projects.

The DdL research project started in 1990 and finished in 1993. This is a theoretical project looking for a uniform logical framework for deductive and abductive diagnosis. Although the proposed framework is independent, we used mainly digital circuits. The main theoretical results are reported in [1].

Unlike the DdL project, the SeD research project is directed towards the methodological level. This project resulted in a prototype, called *sidi*, which was evaluated in cooperation with the iron industry. The main goal of this project was to propose methods, tools, and languages to construct a diagnostic system for large industrial plants. Results are reported in [2].

The NdL research project started in 1992 and is ongoing. It aims at an efficient mechanization of abduction by using neural networks. This project resulted in a neural model that was successfully tested on 150 actual medical cases [3,4]. Our current research is to further extend the model to take into account complex classes of abduction problems from different application areas.

In 1994, the MMS research project was started. This project aims at defining an appropriate diagnostic method for reactive systems. The target applications are mainly in industrial engineering. The completion of the project is scheduled for 1997.

### **References:**

- [1] Ayeb, B., Marquis, P. and Rusinowitch, M. (1993). Preferring Diagnoses by Abduction, *IEEE Transactions on Systems, Man and Cybernetics*, 23(3), pp. 792-808.
- [2] Ayeb, B. (1994). Towards Systematic Construction of Diagnostic Systems for Large Industrial Plants: Methods, Languages and Tools, *IEEE Transactions on Knowledge and Data Engineering*, 6 (5), pp. 698-712.
- [3] Ayeb, B. and Wang, S. Computing effect-to-cause/cause-to-effect diagnoses within NdL, In *Proceedings of IJCAI-93 (Thirteenth International Conference on Artificial Intelligence)*, pp. 1332-1338. Morgan Kaufmann Publishers, 1993.
- [4] Ayeb, B. and Wang, S. (1995). Abduction-Based Diagnosis: A Competition-Based Neural Model Simulating Abductive Reasoning. *Journal of Parallel and Distributed Computing*, 24, pp. 202-212.

Contact: Bechir el. Ayeb, Associate Professor  
E-mail: ayeb@dm.usherb.ca

### **Diagnosis with Markov Decision Processes**

Craig Boutilier

Department of Computer Science  
University of British Columbia

Craig Boutilier is currently using partially observable Markov decision processes to model testing-repair strategies in stochastic processes with competing objectives. Current research involves the use of structured problem representations, problem decomposition, and approximation methods for producing optimal or satisfying test-repair-replace regimes in large problems (the current test domain being a manufacturing assembly process).

Contact: Craig Boutilier

E-mail: ebly@cs.ubc.ca

URL: <http://www.cs.ubc.ca/spider/cebly/craig.html>

### **The APACS Project**

Bryan M. Kramer

University of Toronto

The APACS project is building a system for real-time monitoring and diagnosis of continuous industrial processes using model-based reasoning techniques. A key success factor has been the use of a numerical simulation as a model for identifying discrepancies and checking hypothesized failures. A prototype that diagnoses problems in the feedwater system of a nuclear power generating station has been very successful. The project, which involves the collaboration of Ontario Hydro, the University of Toronto, and CAE Electronics is funded through PRECARN, began in May of 1990 and is scheduled to end in March 1996.

Contact: Bryan M. Kramer, Ph.D. P

URL: <http://www.cs.toronto.edu/~kramer>

### **Neuromuscular Diagnosis Using Multiply Sectioned Bayesian Networks**

University of British Columbia

PAINULIM Project Team

PAINULIM, a prototype neuromuscular diagnostic system based on Bayesian networks, was developed as a research project at the University of British Columbia (UBC) with cooperation from the Neuromuscular Disease Unit of the Vancouver General Hospital (VGH). PAINULIM sets out to diagnose patients suffering from a painful or impaired upper limb due to diseases of the spinal cord and/or the peripheral nervous system. The fourteen most common diseases considered include: amyotrophic lateral sclerosis, Parkinson's disease, anterior horn cell disease, root diseases, plexus lesions, intrinsic cord disease, carpal tunnel syndrome, and medial, ulnar and radial nerve lesions.

The research team was formed by members of UBC and neurologists from VGH. The team members include Dr. Y. Xiang (a Ph.D. student in UBC at the time and on faculty at

U. of Regina at present), Dr. D. Poole (faculty of UBC), Dr. M. Beddoes (faculty of UBC), B. Pant (Neurologist in VGH at the time) and Dr. A. Eisen (Neurologist in VGH).

PAINULIM was based on a Bayesian network that contains 83 variables representing fourteen diseases and 69 features (evidence), each of which has up to three possible outcomes. The network is multiple connected (multiple path exists between a pair of nodes) and has 271 arcs and 6795 probability values. As commonly applied, the Bayesian network representation does not consider domain structure, but rather lumps all variables of a domain into one homogeneous or flat network. This is appropriate for smaller domains. In the development of PAINULIM, such representation was in conflict with the need for (1) efficient computation using the hardware available in hospital laboratories and (2) more natural user interface that allows neurologist to focus their attention on one natural subdomain at a time.

This need inspired the development of a general knowledge representation and inference formalism called multiply sectioned Bayesian networks (MSBNs). A MSBN allows the decomposition of a large problem domain into multiple natural subdomains such that each is represented as a Bayesian subnet. During a diagnostic session, only one subnet is being executed at any one time. This allows the inference computation to be performed efficiently by not having to compute the inactive subdomains. It also presents to the user only the natural subdomain of his/her attention at any time. As the user shifts attention from one subdomain to another, the evidence accumulated in the previous subdomains can be propagated to the newly active subdomain smoothly.

PAINULIM started in 1990 and the prototype was completed in 1992. An evaluation was carried out using 76 patient cases in VGH in comparison with diagnoses made by neurologists. For cases in which neurologists were confident with their diagnoses, PAINULIM's diagnoses were consistent with the human diagnoses. For several difficult cases (multiple diseases may be present), PAINULIM's diagnoses took better account of the evidence and sometimes suggested a second disease not suggested by the human.

#### **References**

##### *On the theory of MSBN:*

Y. Xiang, D. Poole and M.P. Beddoes, Multiply sectioned Bayesian networks and junction forests for large knowledge based systems, *Computational Intelligence* 9 (2) (1993) 171-220.

Y. Xiang, D. Poole and M.P. Beddoes, Exploring localization in Bayesian networks for large expert systems, *Proceedings of the 8th Conference. Uncertainty in AI*, Stanford, CA (1992) 344-351.

##### *On the development and evaluation of PAINULIM:*

Y. Xiang, B. Pant, A. Eisen, M. P. Beddoes and D. Poole, Multiply Sectioned Bayesian Networks for Neuromuscular Diagnosis, *Artificial Intelligence in Medicine*, 5, 293-314, 1993.

Contact: Y. Xiang  
E-mail: yxiang@cs.uregina.ca  
URL: <http://cs.uregina.ca/~yxiang/>

### **Formal Representation and Reasoning in Diagnostic Problem Solving**

*Sheila McIlraith*

Department of Computer Science  
University of Toronto

My doctoral research investigates the formal representation and reasoning issues that underlie diagnostic problem solving. We argue that a comprehensive account of diagnostic problem solving must involve reasoning about action and change. Providing such an account presents at least two fundamental challenges: first, a knowledge representation scheme must be devised to capture the interrelationship between actions and the state and behaviour of a system; second, formal characterization of the tasks involved in diagnostic problem solving must be defined.

We base our claim for the necessity of a theory of action on the following informal observations regarding the nature of physical systems and their associated diagnostic problems. Most physical devices operate within and are affected by the dynamics of their environment. Events occur which alter the state and behaviour of a device. Furthermore, for the majority of devices, testing and repair require the performance of a sequence of actions that may change the state of the device or the world around it. Indeed, diagnostic problem solving is often purposive in nature. A unique diagnosis need not be an end in itself. Identifying candidate diagnoses may only be necessary to the extent that it enables an appropriate action to be selected. Finally, many systems we may wish to diagnose are inherently dynamic. As such, representing the dynamics of their behaviour requires a theory of action and change.

My thesis proposes the situation calculus as a logical language for explicating the representation and reasoning issues involved in diagnostic problem solving. Using this language, we provide a method for incorporating actions into the representation of the behaviour of a system.

Further, we present logical characterizations of diagnosis, testing, and repair for behaviourally static systems whose state can be affected by events that occur in the world, and which require world-altering actions to achieve tests and repairs. Incorporating a theory of action into our formalism provides for a comprehensive account of diagnosis, testing, and repair. Actions are employed both as observations to help *project* what may be wrong with a system, and as diagnoses to *explain* what happened to cause observed behaviour. In diagnostic problem solving, testing is performed to discriminate a space of candidate diagnoses. We examine the formal issues related to testing, including test design, test generation, and the conclusions that can be drawn from the outcome of a test. Achieving tests and repairs may require world-altering actions. Consequently, contemplating their realization is characterized as planning

in the face of competing diagnoses. Finally, we propose the use of high-level procedures for testing, repairing, and reacting. A logic programming language called GOLOG is utilized to define these high-level procedures which are instantiated with respect to a particular system, producing a primitive action sequence within the situation calculus.

Contact: *Sheila McIlraith*,

E-Mail: [mcilraith@cs.toronto.edu](mailto:mcilraith@cs.toronto.edu)

### **Model Based Knowledge Organization**

*Rob Wylie*

Integrated Reasoning Group  
Institute for Information Technology  
National Research Council of Canada

Model Based Knowledge Organization (MBKO) refers to a set of techniques for representing and manipulating knowledge about physical, engineered systems.

The development of MBKO is motivated by the following observations. Many reasoning tasks are dominated by mappings between structural/functional (as in transfer functions) representations of physical systems and behavioural representations. Many different types of structural/functional representations exist, each appropriate to a distinct class of behaviour elicitation task. Often the solution of an engineering problem requires the use of more than one of these representations. Finally, in many engineering problem solving environments, a single physical system is persistently the subject of analysis while the specific reasoning tasks are transient and variable in nature.

To support reasoning in such environments, an information system design should focus on the management of structural/functional and behavioural knowledge about the physical system. It should provide the tools for mapping knowledge between modelling domains. Finally, it should provide the tools to dynamically compose appropriate models of the physical system on demand.

MBKO provides such a framework. It defines a domain independent language for representing models and behaviour of physical systems (called Model Behaviour Pairs or MBPs). It provides a domain independent means of representing reasoning tasks in terms of incomplete MBPs. It provides a consistent mechanism for translating behavioural knowledge from one model (in one domain) to another model (in another domain). Finally, it provides a set of operators for selecting predictive models that are in some sense minimal and adequate in the context of a specific reasoning task.

The MBKO language and operators are based upon a definition of "dependency" which integrates knowledge about the task, the model topology, the system's behaviour, and the modelling domain. It is this abstraction of domain, system, and task knowledge which allows MBKO to consistently handle a diversity of model types.

To support the claim that MBKO is a reasonable representational framework for situated reasoning about physical systems, an implementation has been built and examples run which demonstrate computational and control-

of-reasoning improvements over conventional approaches in the modelling domains of planning, qualitative physics, and discrete event simulation.

Contact: Rob Wylie,

E-mail : [rob@ai.iit.nrc.ca](mailto:rob@ai.iit.nrc.ca)

### **Pegasus Project**

*Bernie Pagurek*

Department of Systems and Computer Engineering  
Carleton University

Since 1990 the Pegasus group at Carleton University (Systems and Computer Engineering) has carried out diagnostic research as part of its investigations into communications network fault management. New probabilistic and logical methods have been created and tested on simulated and operational networks, as well as on the standard UCI suite. In addition, the group has been working on combined reasoning and network protocol management agents to facilitate rapid deployment of experimental diagnostic and DAI algorithms on live networks.

Contact: *Bernie Pagurek*

E-Mail: [bernie@sce.carleton.ca](mailto:bernie@sce.carleton.ca)

### **Software Diagnosis**

*Greg Bond*

Department of Electrical Engineering  
University of British Columbia

Currently we are investigating the application of consistency-based diagnosis to large scale software systems implemented in C and C++. Our goal is to be able to assist with the localization of faulty programming constructs using information gleaned from users, program traces, and core dumps.

Contact: *Greg Bond*

E-mail: [Bond@ee.ubc.ca](mailto:Bond@ee.ubc.ca)

URL: <http://www.ee.ubc.ca/~bond>

### **Laboratory for Computational Intelligence**

*David Poole*

Department of Computer Science  
University of British Columbia

Craig Boutilier, Alan Mackworth, and David Poole of the Laboratory for Computational Intelligence at UBC have been working on the theory and practice of diagnosis from both a logical and probabilistic perspective, including logical characterizations of diagnosis, combining logic and probability for diagnosis, Bayesian networks for diagnosis of medical and engineering domains, qualitative evidential reasoning, modelling dynamical systems, and decision making under uncertainty.

Contact: *David Poole*

E-mail: [poole@cs@ubc.ca](mailto:poole@cs@ubc.ca)

URL: <http://www.cs.ubc.ca/spider/poole>

### **The Diagnostic Remodeler**

*Suhayya Abu-Hakima*

Seamless Personal Information Networking (SPIN)  
Institute for Information Technology  
National Research Council of Canada

My thesis research work addressed the problem of automated model acquisition through the re-use of fault knowledge. The Diagnostic Remodeler (DR) algorithm was implemented for the automated generation of behavioural component models with an explicit representation of function through the re-use of fault-based knowledge. DR re-uses as its first application the fault knowledge of the Jet Engine Troubleshooting Assistant (JETA). DR extracts a model of the Main Fuel System using real-world engine fault knowledge and two types of background knowledge as input: device dependent and device independent background knowledge. To demonstrate DR's generality, it was also applied to a coffee maker fault knowledge base to extract the component models of a full coffee device. The work was started in 1992 and completed in 1994. The DR generated engine model was validated by an NRC engine expert.

Contact: *Sue Abu-Hakima, Group Leader, SPIN*

E-mail: [suhayya@ai.iit.nrc.ca](mailto:suhayya@ai.iit.nrc.ca)

### **Equipment Diagnosis**

*Jeff Xi*

Institute for Sensor and Control Technology  
National Research Council of Canada

Diagnostics is one of the technical areas in the Institute for Sensor and Control Technology (ISCT) of the National Research Council Canada. The main focus on development of measurement and analysis techniques for detecting and isolating faults in complex machinery and processes. Technologies that have been, or are being developed at ISCT, include vibration and acoustics, and image analysis. These technologies have applications in Canadian industries such as mining, forestry, wood products, manufacturing, transportation, and aerospace.

Research in the area of vibration is focused on advanced signal processing techniques for extracting useful information from non-stationary and non-linear signals. Methods under study include the short-time Fourier transform, Winger distribution, wavelets, holospectrum, qualitative dynamics, amplitude and phase demodulation. Analytical and experimental investigations have been carried out for diagnosing gear and bearing failures, rubbing, cracking, and stick-slip friction-induced vibration. Some of the technologies developed are being introduced to industry through projects, such as diagnosis of helicopter gear boxes, vibration condition monitoring of sawmills, and vibration condition monitoring of injection molds.

The research focus in the acoustics area is on machinery noise source identification. Novel noise identification

*Continued, page 12*



# Diagnostics R&D in the Integrated Reasoning Group at the National Research Council's Institute for Information Technology

Mike Halasz

## Introduction

On présente dans cet article les projets en cours dans le Groupe de raisonnement intégré (RI) de l'Institut de technologie de l'information du CNRC. Le groupe a été formé en avril 1995 suite à la division du Laboratoire des systèmes de connaissances du CNRC en groupes ayant des objectifs technologiques plus cohérents. On y fait de la recherche et du développement sur l'application de l'intelligence artificielle aux processus de prise de décision où le temps est un facteur critique, dans des environnements où il y a une surabondance de données et d'information utiles. Habituellement, ces données et cette information sont distribuées, comportent du bruit et se retrouvent sous des formes diverses. L'expertise sur la manière d'utiliser l'information est éparpillée entre différentes personnes, documents, techniques d'opération et logiciels. Le travail de recherche appliquée inclut le développement et le raffinement de techniques de raisonnement spécifiques et des architectures des systèmes sous-jacents. Les techniques de raisonnement comprennent la logique floue, le raisonnement basé sur des cas, les réseaux de décision, l'induction et le raisonnement basé sur un modèle. Les architectures des systèmes ont des caractéristiques qui permettent l'intégration de sources d'information diversifiées et l'utilisation de techniques de raisonnement multiples. Des prototypes fonctionnels illustrant des concepts et explorant des possibilités commerciales sont développés avec des utilisateurs de première ligne et des développeurs de logiciel. Nous avons pour objectifs d'identifier et d'éprouver de nouveaux produits et services pour nos partenaires tout en demeurant pertinents sur le plan de la recherche. Nos applications actuelles touchent le diagnostic et la gestion de la réparation d'équipement complexe.

## Introduction

This article discusses the ongoing projects<sup>1</sup> of the Integrated Reasoning Group (IR) at the NRC's Institute for Information Technology. The group was formed in April 1995 when the NRC's Knowledge Systems Laboratory was divided into groups with more coherent technological foci. The group does research and development on the application of artificial intelligence to time critical decision making processes in environments where there is an overabundance of useful data and information. The data and information are usually distributed, noisy, and represented in diverse forms. Knowledge of how to use the information is spread out amongst humans, documents, operating practices, and

software. Being application driven, the research effort includes the development and refinement of specific reasoning techniques and the underlying system architectures. Reasoning techniques include fuzzy logic, case-based reasoning, decision networks, induction, and model-based reasoning. The system architectures have characteristics which support integration of diverse information sources and use of multiple reasoning techniques. Functional prototypes, which illustrate concepts and explore market opportunities, are developed with lead users and software developers. Our objectives are to identify and prove new product ideas and services for our partners while maintaining our research relevance. Current applications deal with diagnosis and the repair management of complex equipment. The reasons for this focus are:

- a need for the technology.<sup>2</sup> For every dollar spent on new machinery, 51 cents are also spent on maintaining existing equipment. This amounts to repair costs of approximately \$15.3 Billion per year in Canada alone.<sup>3</sup>
- collectively, the group has an appropriate blend of engineering and computer science backgrounds to tackle these types of problems.
- diagnosis is a challenging area of AI research.

## Projects

### Paper Maker's Advisor (PMA)

PMA is a follow-up to an earlier project described in a paper presented at the 81st (1995) CPPA Technical Section Meeting of the Canadian Pulp and Paper Association (CPPA) [1]. That paper describes the Paper Drying Expert System (PDES), a prototype expert system to assist machine operators diagnose equipment failures and sub-optimal operation. It was validated with tests in two paper mills. The results were sufficiently positive to warrant further development towards on-line implementation and commercialization.

The PMA project is expected to start in the first quarter of 1996. Whereas the PDES prototype was off-line, PMA will have the flexibility to accommodate various amounts of on-line data (depending on available instrumentation in a given mill); the remaining data will be entered manually.

PDES is configurable. Tailoring for the different equipment types found in practice is straightforward. A major output of the PDES prototype is to inform the user when the steam and condensate system is wasting energy and to quantify thermodynamic performance. It achieves this by comparing the actual amount of condensate and the current U-factor

with established standards for the specific machine. If the machine is operating outside of specified thresholds, the expert system enters into a diagnostic dialog with the user to obtain more information aimed at determining the possible cause(s) of the deteriorated performance. Some of the possible faults are: (1) steam supply problem, (2) wet end water removal problem, (3) control loop problems such as steam pressure, differential pressure, or water level, (4) incorrect pressure or DP set points, (5) defective components such as thermocompressors, vacuum pumps, condensate pumps, and syphons, and (6) felt tension problems.

PMA will extend the scope of PDES to include other parts of the paper machine, e.g., ventilation and press sections. Research and development will also aim to: (a) enhance the diagnostic mechanism to include better diagnostic explanations and the ability to deal with incomplete evidence, (b) provide a knowledge base maintenance interface, and (c) achieve performance improvements by data analysis and machine learning once PMA is installed. Items (a) and (b) will take approximately one year, whereas item (c) will be done in the following year.

NRC's partners in the PMA project are Abitibi-Price and Kruger which are mill operators providing test sites, KanEng which will provide domain expertise, and Proconex which will develop and support the resulting product. NRC will focus mainly on research aspects. The partners are all members of SIMCON (Software for Integrated Manufacturing Consortium), an umbrella organization that provides management services for collaborative projects.

## Reference

[1] R. Amyot, J. Gowing, R. Wylie, R. Henzell, J. Futcher, J. Reinsborough, A. Coderre, P. Henzell, and O. Vadas, "Steam and Condensate Diagnostic Expert Systems", 81st Annual Meeting, Technical Section, Canadian Pulp and Paper Association, Montreal, Jan. 31 - Feb. 3, 1995, pp. A41-A44.

## Integrated Diagnostic System (IDS)

### Background

IDS originated a few years ago when NRC approached a number of manufacturers and operators of aircraft engines with a proposal to co-develop a hybrid diagnostic system. As a result of this initiative, a group was formed which consisted of General Electric (including GE Aircraft Engines USA, GE Corporate Research and Development (GE CRD), and GE Canada, Air Canada, and NRC). After an extensive requirements and benefits analysis carried out at Air Canada, the scope of the project was extended beyond aircraft engines to ultimately include all aircraft systems on all fleets.

The study at Air Canada also revealed the following maintenance issues:

- There is a need for systems which help in time-critical, accurate, and consistent decision making.
- Technicians must deal with many more types of equipment than ever before. Access to the right data, knowledge,

and expertise at the right time is necessary to effectively carry out the maintenance mission.

- Newer generation equipment is producing increasing amounts of potentially useful data. Tools are required to aid in the integration and interpretation of data.
- Unscheduled maintenance is a persistent problem.
- Significant savings could be realized through reduction in "No Fault Found" and repeat snag occurrences.
- To compete effectively, the airline must continue to seek practical cost cutting solutions. The pressure of "doing more with less" is prevalent.

These concerns pointed to an IDS concept that would provide the following benefits and capabilities:

- enhance maintenance performance levels at all sites,
- improve diagnostic accuracy and reduce ambiguity in fault isolation,
- advise on real-time repair action,
- provide clues to incipient failures, and
- access and display relevant maintenance information.

We have carried out other studies and have found that the problems encountered by aircraft operators are similar to those being experienced by other fleet operators, such as trucking firms, railways, and mining companies. Thus, we feel that IDS can extend to these areas.

### Description of Decision Making Environment

The main objective of IDS is to help the technician isolate faults. This involves examining symptoms, automatically assembling relevant data, and from this, determining what the most likely cause of the fault is.

Figure 1 depicts the world within which IDS must operate. Aircraft are continually on the move and turn-times at the gate are increasingly shorter to maximize utilization. Modern aircraft, such as the Airbus A320, have systems on board which transmit data to ground stations. These data consist of routine performance snapshots (e.g., altitude, ambient temperature, hydraulic pressures, engine temperatures, valve positions, etc.), pilot messages, aircraft generated fault messages, and special purpose reports which are generated when prescribed limits are exceeded.

There are a number of databases which support maintenance in addition to the one described above. Some of the ones more relevant to maintenance contain descriptions of symptoms and the associated maintenance actions (free form text), deferred problems, component reliability, and parts location. There is also a wealth of relevant information held at the manufacturer, and by people and information systems in the engineering and maintenance control departments. This is not available to the line technician in a timely manner.

When aircraft problems occur, complications can arise. The technician's objective is to avoid delays while keeping the aircraft airworthy. In addition to their professional judgment and knowledge, current tools at the line technician's disposal are:

- the aircraft Built-In Test Equipment (BITE)

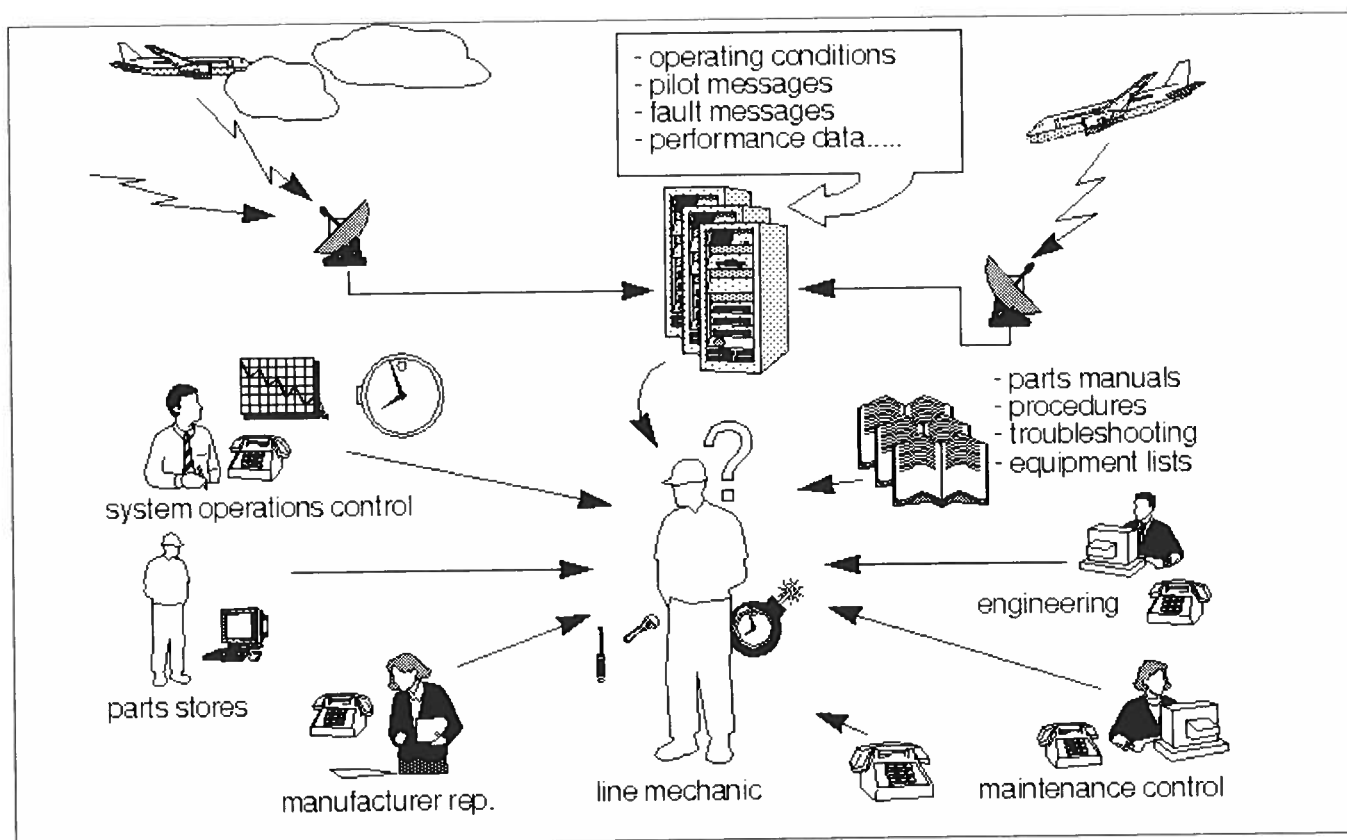


Figure 1. Current Situation - airline

- the Minimum Equipment List (MEL) (a legal document found on board all aircraft defining minimum equipment requirements for dispatch safety)
- the aircraft log book which provides free text descriptions of the problems encountered
- post-flight reports (a summary of the messages generated by the aircraft for a given leg)
- troubleshooting manuals (currently on microfiche but moving towards CD-ROM)

Due to the many distributed information and knowledge sources, and the importance of making the right decisions, this environment represents a perfect example of the type of problems the group is interested in tackling.

### IDS Functions

To meet airline maintenance requirements, IDS must support three functions:

#### IDS-1: "Provide Accurate Diagnosis"

This is the basic building block lying at the heart of IDS. It determines "root-causes" of fault indications generated either by onboard equipment or the flight and maintenance crews.

This function requires both interpretation and validation of faults generated by the aircraft systems. Central to the operation of IDS-1 is an "Event Monitor" or "Intelligent Filter" which screens out false alarms and spurious messages. The generation of thousands of such false messages in a

typical month of airline operations is a significant problem because they mask the real problems. The result: possible false removals or unnecessary maintenance actions. The "event monitor" considers the context within which fault messages occur (i.e., flight regime, corroboration from other parameters/ subsystems, etc.) to validate the fault.

Once the fault has been validated, IDS-1 develops a diagnosis by using automatically generated rules extracted from the troubleshooting manual and matching symptom patterns (or "clusters")<sup>4</sup> to previous examples or cases. The intent is to recommend a solution to the problem as soon as possible. In cases where the technician is forewarned early enough, various preparation activities can be done outside the time critical "turnaround" interval.

#### IDS-2: "Advise Optimal Repair Strategy"

IDS-2 takes the fault diagnosis provided by IDS-1 and considers situational criteria to recommend the best course of action to "manage" the fault. Examples of these criteria are: parts availability, Minimum Equipment List (MEL), impact, aircraft routing, weather, turn time at the gate, labour availability, deferred "snags," and "check" schedules.

Recommendations generated by IDS-2 range from identifying parts which should be replaced at the gate to delaying action to a downtime station or night layover (provided the aircraft can be dispatched according to the MEL).

### **IDS-3: "Assess Equipment Health/Predict Incipient Failures"**

The function of IDS-3 is to identify incipient equipment failures and provide a basis for on-condition maintenance. It will predict failures before they occur rather than troubleshooting after the failure occurs (which is the function of IDS-1) and will support maintenance planning.

IDS-3 will utilize innovative techniques to identify patterns and trends in data. For engines, the data can include snag histories, indicated minor "events," performance data, trends in maintenance activity (e.g., oil consumption), vibrational signatures, maintenance observations, time-limited parts life tracking, etc. The key here is that although individual pieces of information may appear insignificant, when a composite picture is made of the situation and patterns and/or trends in that composite picture are made over time, more accurate predictions are possible. These predictions can then be fed back to IDS-2 for planning and scheduling purposes.

### **Prototype Status**

#### **General**

Development of the prototype started early in the spring of 1995. It is focused on the engine and flight control subsystems of the A320. These were selected in order to illustrate the proof of principle on the most problematic subsystems. The methodology extends to other aircraft subsystems. The development environment selected is ART\*Enterprise which dictated the use of Pentium workstations running Windows 3.1.<sup>5</sup> The team specifying and building the prototype consists of six persons.

#### **Information Sources**

Two of Air Canada's databases have been continually (since October 1994) read into a DB2 database running on one of the workstations (server) in OS/2. The two databases are of prime importance to maintenance of the aircraft. They are:

- AIMS (Aircraft Information Monitoring System), the database which stores all the aircraft and pilot generated messages and data. These messages serve as the raw symptom data which IDS-1 interprets.
- MAS (Maintenance Automation System), which provides free form text descriptions of the symptoms and the repair carried out. This data is entered by technicians and pilots.

#### **Information Processing**

The data can be "played back" from starting at any chosen date in its history - exactly as it happened, including incomplete, noisy, and extraneous messages. Each incoming message is processed according to the following criteria:

1. Is the message a known nuisance that can be ignored?
2. Is the message a known nuisance that can be ignored under certain conditions?
3. Can the message be grouped together with others in a logical manner? Here things like time of occurrence, relationship of the reported subsystems, and system logic (as governed by the rules extracted from the

troubleshooting manuals) are taken into account.

4. Is there a pilot-generated message that can be matched with another potential "no go" situation upon arrival at the downline station?
5. Is the message part of an ongoing problem that has persisted for a number of days?

Each of these messages eventually forms a grouping and can evolve as more information is produced. The groupings are used to automatically retrieve the ambiguity group (the list of suspected components) as governed by the troubleshooting manual and also similar examples stored in a case-base. The case-base is currently generated in a two step process. The first step automatically attaches a message grouping to a repair action described in MAS by extracting clues from the free-form text description of the snag. The resulting object (now containing both the symptoms and the fix) is stored in a database which constitutes the repair history of the aircraft or fleet. In the second step, the objects are presented to an experienced technician who decides whether or not they should be moved into the case-base.

#### **Interface**

There are three main screens that the technicians typically use. They are the Fault Event Monitoring, Fault Resolution, and Snag Rectification screens. A brief description follows:

##### **Fault Event Monitoring Screen**

The Fault Monitoring Screen shows the fleet fault status. The aircraft are identified by their identification numbers and their current position is shown (i.e., origin, enroute, or destination). Beneath each identification number are icons for the aircraft's four major subsystems: air frame (AF), avionics (AV), engines (ENG), and auxiliary power unit (APU). These icons and those for the aircraft change colour depending upon the reasoning of the system. Currently, there are three colours:

- grey = normal (there can still be some problems detected but they do not have any bearing on safety)
- yellow = a possible no go situation upon arrival (not verified by the pilot)
- red = a possible no go situation upon arrival (verified by the pilot)

##### **Fault Resolution Screen**

This screen provides the technician with a recommendation as to the probable cause of the fault. It is divided into three sections: a grouping of symptoms, the ambiguity group identified in the troubleshooting manual, and the most probable causes based on past experience (case-base). It should also be noted that IDS will process all the faults generated by the aircraft and will generate recommendations for each one, provided they are distinct. From this screen the technician can automatically access the relevant pages of the Minimum Equipment List (in HTML, using Netscape) and the electronic parts and troubleshooting manuals (both are on CD-ROM).

##### **Snag Rectification/ History Screen**

If the technicians wish to do some of their own troubleshooting, they can access a Snag Rectification/History

Screen which provides a history of all maintenance relevant to the current problem. This feature is found to be very useful for the technicians and engineers since they all attempt to maintain their own lists of interesting repairs to guide their actions.

### Future Directions

IDS is an ambitious project with broad scope. The prototype, along with its infrastructure and data resources provides an excellent platform upon which to test new ideas. It recently underwent a one week evaluation by twelve Air Canada technicians and engineers and received very positive reviews. Immediate plans are to connect IDS to "live" data streams (replacing the captive AIMS and MAS databases) and deliver it to selected Air Canada personnel for extended testing. The users will be experienced A320 fleet troubleshooters and will provide feedback on system accuracy and performance. We feel that the prototype demonstrates technical and market viability of a product and are currently negotiating with companies that would deliver and support subsequent versions of IDS.

Longer range plans are to:

- explore more refined ways in which to extract meanings from free form text;
- improve upon the symptom clustering strategies and the indexing to the case-base;
- investigate the feasibility of automatic casebase creation and maintenance from AIMS and MAS data;
- integrate the GE Aircraft Engine performance software and experiment with model-based trend analysis;
- continue experiments with automation from aircraft performance data to provide richer symptom sets for case retrieval. We are currently investigating knowledge driven constructive induction, data driven constructive induction, and automatic trend recognition;
- develop and integrate other reasoning techniques (for both IDS-1 and histories, aircraft repair and maintenance histories, deferred problems, parts location, and flight movement (including weather).

### Diagnosis Research in Canada (con't. from page 7)

procedures have been developed and integrated into the ISO standards. Ongoing industrial projects include acoustic monitoring of railway bearings, and acoustic monitoring of plasma spray tool wear.

Research in the area of image analysis is aimed at application of the image analysis technique to fault detection, in cooperation with industrial companies. One technology under development is an intelligent debris analyzer, to identify types of wear materials and wear condition of machinery. Another technology is a sensing system for lumber manufacturing, to locate log end defects in order to optimize cuts to maximize the product values.

Contact: Jeff Xi

E-Mail: xi@ai.iit.nrc.ca

•extend the concept to other aircraft types such as the Airbus 319/340 and Boeing 767/747.

### Other Projects

In addition to the above, the group is actively pursuing initiatives in the automated medical laboratory, underground mining, and manufacturing fields.

### Notes


1. The projects underway with clients are covered. See our web site for descriptions of internal and past projects.

URL: [http://ai.iit.nrc.ca/IR\\_public/](http://ai.iit.nrc.ca/IR_public/)

2. Source: *Private and Public Investment in Canada*. Statistics Canada, Investment and Capital Stock Division, Revised Intentions 1990, catalogue 61-206 Annual.

3. This was calculated for ten industrial sectors only. The estimate would be considerably higher if all Canadian industrial sectors were taken into account.


4. Currently these clusters are formed by encoding high level "analysis" rules that experienced technicians find useful when interpreting aircraft and pilot-generated fault messages. Experiments are underway to create richer symptom sets (e.g., including measured parameters such as temperature or pressure) through the use of induction algorithms.

5. ART\*Enterprise runs under the OS/2 operating system and now on various Unix workstations. Of the three versions, the Windows version was the most mature. 

---

*Michael Halasz has been with NRC's Institute for Information Technology since 1988. He is currently leader of the Integrated Reasoning Group. He has previously held positions in the Canada Post Corporation as a project engineer for network design and optimization, and the General Electric Company as a field engineer for industrial steam and gas turbine service and installation. He received a Bachelor's degree in Mechanical Engineering from McGill University (1980) and a Master's in Systems Design Engineering from the University of Waterloo (1985).*

---

*Sue Abu-Hakima is the Group Leader of the Seamless Personal Information Networking (SPIN) Group at the Institute of Information Technology at NRC. She has been a researcher at NRC since 1987. She had previously worked at BNR starting in 1982. She has Doctorate and Masters degrees in AI from Carleton University's Department of Systems and Computer Engineering. She also has a B. Eng. from McGill University's Department of Electrical Engineering with specialties in Communications and Computers. She is on the Editorial Board of the International Journal of Human-Computer Studies.* 



# PUMP - An Integrated Maintenance Scheduling and Costing System

*Steve Reilly*

## Résumé

Le système d'entretien PUMP constitue le développement le plus récent d'un programme de plusieurs années qui vise à concevoir, implanter et vérifier une série de systèmes pour la surveillance de la condition et de la performance d'équipement rotatif et alternatif. PUMP est un système intégré de planification et d'évaluation des coûts de l'entretien qui reçoit des données de sous-systèmes analysant les vibrations, l'huile de lubrification et la performance, de même que d'autres aussi d'autres systèmes et des ingénieurs. PUMP utilise ces différentes sources de données pour poser des actions d'entretien, pour résoudre des conflits possibles dans l'information provenant des sous-systèmes, pour suggérer des options de planification, pour générer des commandes de travail et pour archiver les détails de l'entretien. Encore plus important est le fait que PUMP génère des rapports d'entretien conçus pour faciliter la prise de décision par l'équipe d'entretien.

## Abstract

The PUMP maintenance system is the latest development in a multiyear program to design, implement, and field test a series of systems to monitor the condition and performance of rotating and reciprocating equipment. PUMP is an integrated maintenance scheduling and costing system that accepts inputs from vibration, lubricating oil, and performance analysis subsystems, as well as other ship systems and engineering staff. PUMP uses these various inputs to provide maintenance actions, resolve conflicting information that may arise from the subsystems, suggest scheduling options, generate work orders, and archive maintenance history. Most importantly, PUMP generates maintenance reports that are designed to ease the decision-making process for maintenance staff.

Section One describes the goals of the maintenance project as a whole. The functionality of all of the maintenance advisory systems developed during the complete project and PUMP's ability to access and integrate the recommendations produced by these systems, are detailed.

Section Two covers the maintenance management functions contained within the PUMP system. PUMP uses an intelligent scheduling algorithm for maintenance work orders. This allows the maintenance engineer to develop a maintenance plan based on any type of scheduling trigger (i.e., running/calendar hours, sensor alarm levels, expert system recommendations, and ship operating state). The methodology used to produce these maintenance schedules is a rule-based expert system, but this underlying complexity is hidden from the maintenance engineer by an easy-to-use Graphical User Interface.

Section Three covers the reporting and query capabilities of the system. The maintenance engineer can custom design reports and work order forms for many different tasks. Customized work order forms can be specifically tailored for widely differing jobs such as watch-keeping tasks, on-ship maintenance operations, or major refit jobs with substantial contractor labour requirements. A powerful query facility can be used with the form system to produce comprehensive reports.

Section Four details the benefits of the system to the end user. Specifically, it describes the facilitation of the maintenance scheduling function on the ship, and the intention of shipboard staff to use the information generated by PUMP as a major tool to justify extensions in survey periods.

## 1. Introduction

The application of various predictive maintenance techniques has expanded greatly over the last ten years. Such tools as vibration analysis, oil analysis, ultrasonics, and thermography are now commonplace in many types of industrial plants. At the same time, maintenance management systems have been refined into highly developed cost capture and tracking systems for asset management. Unfortunately, the two disciplines have not had a substantial level of overlap, and many of the benefits of both predictive and preventive maintenance are lost. Because the information found in the predictive maintenance tools is not accessible for maintenance scheduling and cost tracking, it is difficult for the maintenance planner to effectively use this information when determining work schedules and equipment replacement planning. Conversely, the equipment work order history contained in maintenance management systems is highly valuable when performing condition assessment using predictive maintenance tools, yet this information is often difficult to access in a useful manner for the predictive maintenance specialist.

PUMP seeks to integrate existing maintenance systems and streamline the workload of maintenance crews by providing a focal point for decision making and offering scheduling functions that allow all maintenance options to be fully explored. The goal of the program is to draw the most benefit possible from the predictive tool-kit in place, and minimize activities that require rigid time-based scheduling for tear down and inspection.

In 1986, a project was initiated by the Transport Development Centre and the Canadian Coast Guard to develop a series of maintenance tools usable by shipboard personnel. PUMP is the last of the tools to be developed

under this project. Tools developed previously in the project have been designed to facilitate condition monitoring on board Canadian Coast Guard ships. The Vibration Analysis for Rotating Machinery INTERNALS project (VARMINT) produced a system which accepts raw vibration signals taken from vibration sensors and produces an analysis of the condition of the machine. This system has been both a technical and commercial success, and is being used in over 300 vibration analysis programs worldwide. Recently, this system was enhanced to allow the joint analysis of vibration and lubrication data. Lubricants can be analyzed by commercial testing laboratories, or by on-ship lubricant analysis instruments, to track particle counts, viscosity, wear metals, contaminants, etc. These data are trended in the DMSI LubriScan system, and the analysis results are made available to PUMP through the VARMINT data analysis and advisory system. VARMINT is better known in the condition monitoring marketplace as Prism<sup>4</sup> Pro.

Performance parameters that are monitored in real-time can also be integrated through the Performance Analysis Software System advisory program (PASS). This system is currently being evaluated aboard one Canadian Coast Guard ship. There is continuous verification of temperatures, pressures, flows, and other critical parameters that are compared against expected baseline data with vigilant monitoring against alarm thresholds. These alarms can be tailored to account for equipment loading and ambient conditions. The operator can request a further analysis of the data through a rule base. The results of this analysis are made available to PUMP.

## 2. PUMP Features

The user is presented with several options regarding the style of PUMP user interface to suit both personnel preference and experience level. The standard windows pull-down menu interface is provided, or the user may choose to activate the toolbar with its self-explanatory icons (with tool tips) or as a third option, the ship schematic access routines can be activated where a drawing/image of the ship is displayed with the various equipment sets positioned as one would expect (refer to figure 1).

The applications menu, when pulled down will display buttons to activate vibration systems, LubriScan and VARMINT. PUMP functionality is organized by a series of data managers. These can be accessed either via icon or pull-down menu and they are segmented into equipment, tasks, personnel contractors, and parts managers.

The process menu activates the scheduler which is an intelligent process that takes into account time-based, condition-based, and consumption-based requirements (among others—see section two) when producing Gantt chart styled scheduling options to the user. One can also activate observation sets (tracking operational status, weather forecasts, seasons, etc.) and investigate machine usage data either tracked automatically by PASS or from manual input. Resource leveling options can be investigated to ensure the best mix of personnel are utilized for upcoming activities (much like resource leveling functions found in project management systems).

A complete report generation system is provided where both primary and secondary work orders can be generated

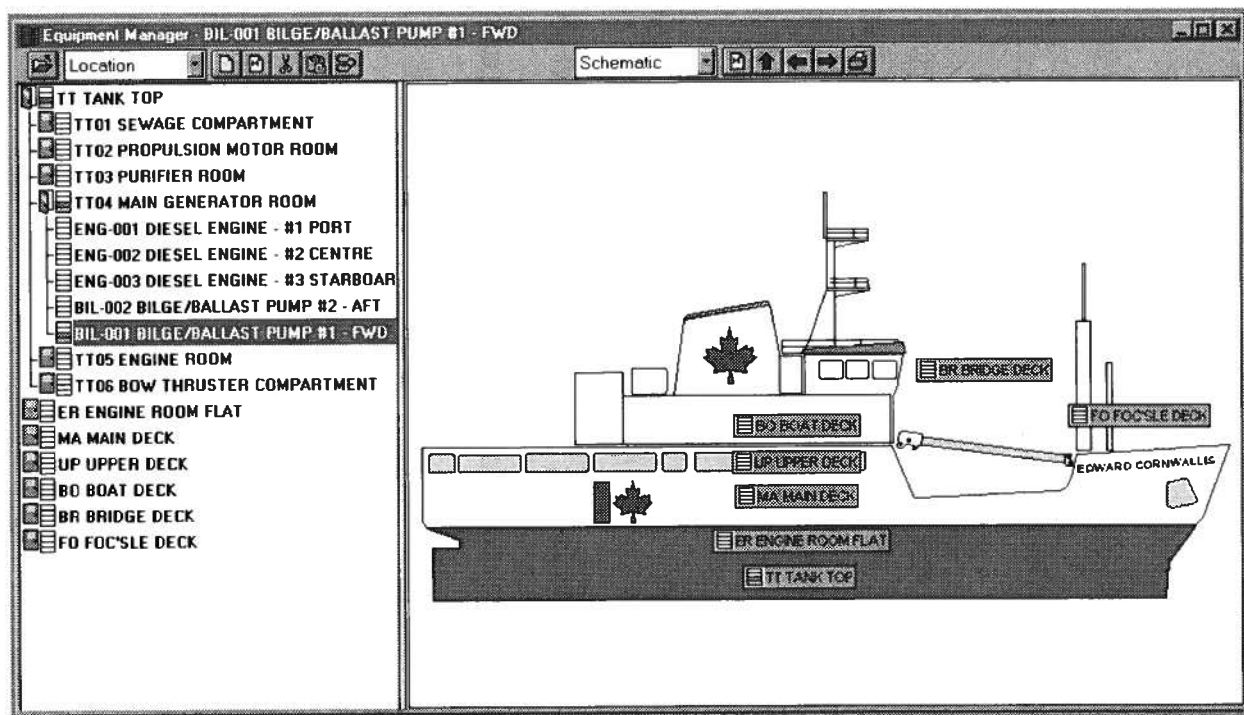


Figure 1. Select From Hierarchy/Select from Schematic

and previewed (refer to section three). Primary work orders have bar codes incorporated to facilitate the closing and tracking of work orders.

A complete set of work order history is maintained, with extensive searching and cross-referencing capabilities. A work order can be accessed by equipment type, task type, personnel on the job, parts used, or contractors used. A complete SQL query facility allows for even greater flexibility in search parameters. Context-sensitive help is available as is a full set of user documentation.

### Advantages of Integration

PUMP users can either enter or import machine running hours, machine usage data, personnel work hours, and contractor charges. Diagnostics from VARMINT (vibration, lubricant analysis, process parameters, etc.) are automatically accessed for the modification of scheduling options. In this manner, conflicting advisories from subsystems can be dealt with and more powerful troubleshooting options can be presented to the maintainer. Convincing reports can easily be prepared as PUMP can import spectral data, lubricating oil histograms, schematics and photographic evidence for inspection authorities to peruse.

### 3. Dynamic Maintenance Scheduling

The scheduling subsystem is the heart of the PUMP system. In standard maintenance management systems, like those commonly available for purchase, the concept of scheduling

is *static*. Once a maintenance schedule has been established, it is fixed. It does not reflect changes in the condition of the equipment or the operating state of the vessel.

Calendar time, running hours, meter levels, etc. are measured, and a maintenance task for a piece of equipment is scheduled based solely on elapsed time, elapsed running hours, and/or reaching a certain number of kilometers, etc.

The PUMP system has the ability to schedule work orders based on calendar time, elapsed time, meters, etc. just as standard maintenance management software does. But the maintenance engineer can also specify actions based on *condition, opportunity, or operation*, taking into account situations that may or may not arise (such as condition monitoring events) and situations whose time frames are difficult to accurately predict (such as ship's operating conditions).

*Dynamic maintenance scheduling* involves using the results found by onboard condition monitoring and the ship's operational state to activate tasks. This is made possible by using an inference engine as the scheduler, rather than a simple algorithmic process. This greatly increases the ability of the operator to tailor a set of scheduling rules that can respond automatically to changing conditions, either in ship operation or machine condition. Instead of simply specifying that a bearing is to be changed every 4500 hours, it is possible to define that the task "Change bearing" on a piece of equipment is to be activated only if the bearing's condition is poor:

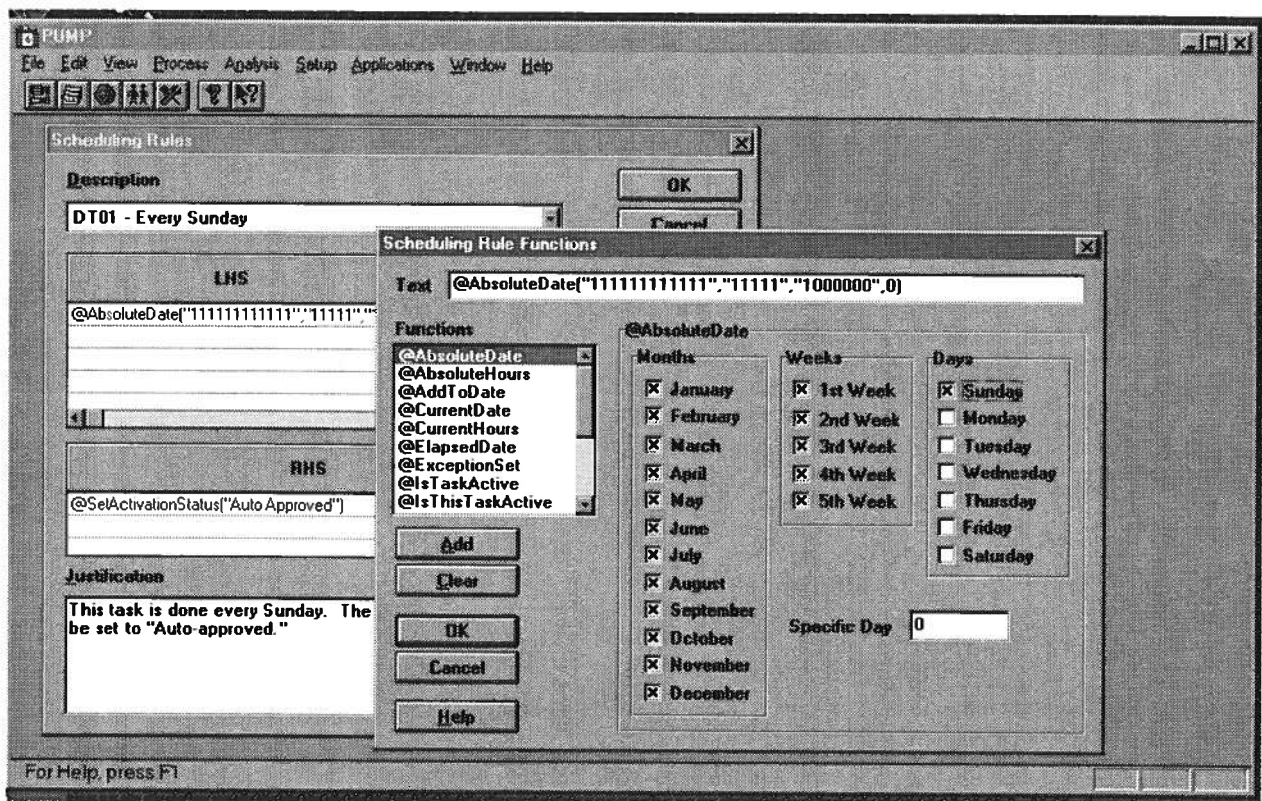


Figure 2. Scheduling Rule Editor

**IF** bearing has been in service for 4500 hours or more,  
**AND** a Phase II or Phase III bearing condition, with severity of HIGH or SEVERE, has been specified by VARMINT.  
**THEN** Change bearing

This allows an in-service meter *and* a condition-based diagnosis to automatically activate a task.

Another possibility—instead of “Clean and adjust injectors” every 1000 hours, the task “Clean and adjust injectors” on the diesel engines can be activated by VARMINT.

**IF** Injectors have not been touched for at least 250 hours,  
**AND** “a High Manifold Exhaust Temperature” has been specified by VARMINT.  
**THEN** Clean and adjust injectors

The PUMP system has a Scheduling Rule Editor which allows the operator to create these scheduling criteria. These rules can range from simple to extremely complicated, allowing the operator to reflect in his system the realities of maintenance scheduling complexity. The Scheduling Rule Editor is shown in figure 2.

What do we mean when we say “activate a task?” Isn’t that the same as “produce a work order?” Well, no. When a

task is activated, it is placed in the Activated Task Queue. It is brought to the attention of the engineer responsible for work order generation, either the chief or the senior. The PUMP system ranks all tasks in order of importance depending on the criticality of the machine, the availability of personnel, parts, etc., and the *operating state of the vessel*. In figure 3, you can see various operating state criteria and active tasks displayed on the scheduling Gantt chart. These operating states are all user-defined. This allows the maintenance engineer to schedule tasks based on opportunity. For example, on a fresh water pump, “Overhaul Motor” is to be activated

**IF** Cracked or Broken Rotor Bars condition, with a severity of MODERATE or greater, has been specified by VARMINT,  
**AND** a Motor Efficiency fault has been specified by VARMINT.  
**AND** Ship Operating Status is LAYUP in less than four weeks.  
**THEN** Overhaul Motor

**IF** Motor has been in service for 4500 hours or more,  
**AND** Ship Operating Status is LAYUP in less than four weeks.  
**THEN** Overhaul Motor

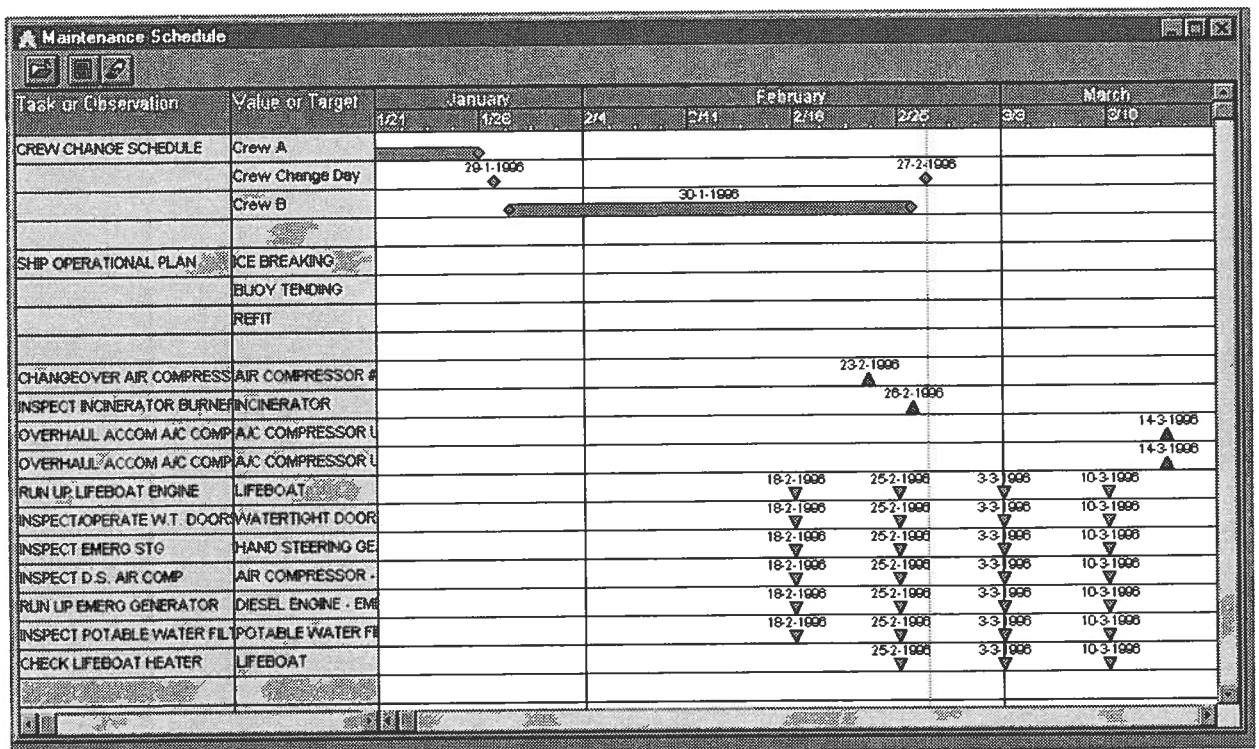


Figure 3. Maintenance Scheduling Gantt Chart

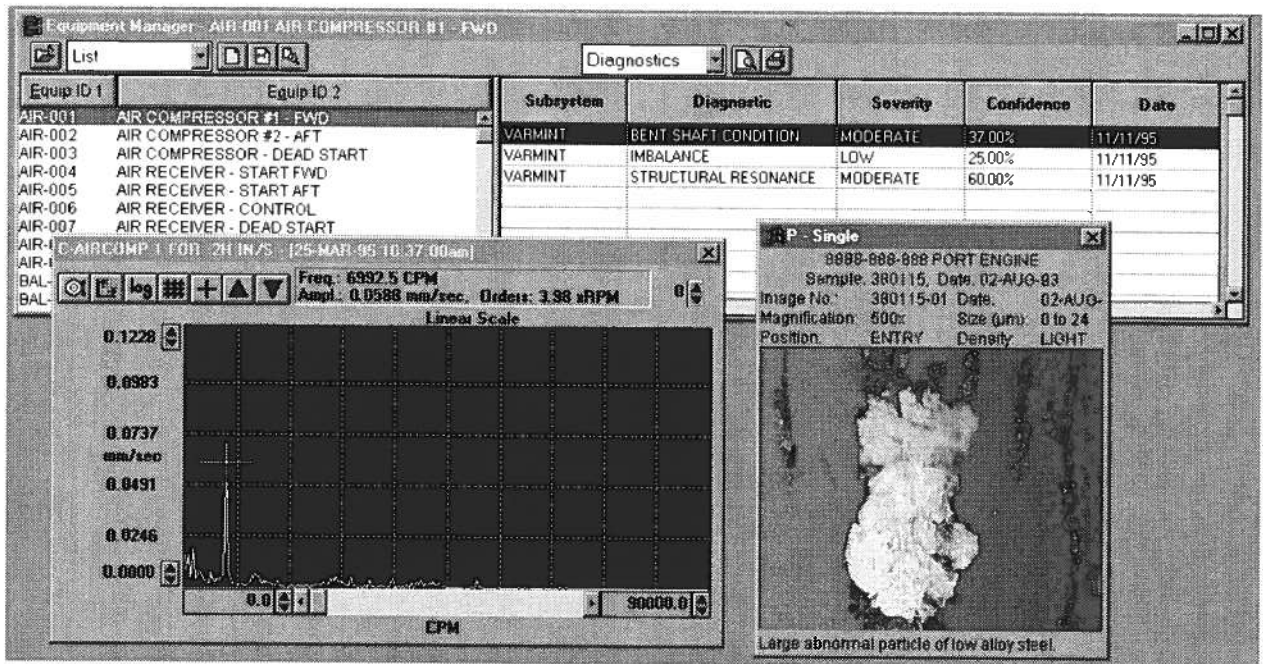


Figure 4. Condition Based Diagnostics

We are allowing the system to schedule an overhaul of the fresh water pump if it has been found in poor condition by VARMINT, or if it has passed its service life, and only if we are within four weeks of a ship operating state that allows us to perform the overhaul. Now, we can take this opportunity to do some additional work.

**IF "Overhaul Motor" task is activated.**  
**THEN Inspect pump impeller, flanges, and piping**

What we are saying here is that a task to check out the condition of the pump should be activated if we are stripping down the motor. Therefore, if the motor is scheduled to be overhauled, the pump is scheduled to be inspected.

Any task placed into the Activated Task Queue can be manipulated by the maintenance engineer. Any activated task can be postponed, canceled, or made higher or lower in priority. Emergency work orders can be manually created by the maintenance engineer, even if they are not being triggered by time or condition. This increases the flexibility of the system. Individual tasks can be defined to either require approval or to automatically generate work orders, reducing the administrative load for the maintenance engineer.

The types of task scheduling that can be performed are:

#### Calendar-based

Calendar-based work order scheduling can use either fixed date scheduling or elapsed time scheduling.

- Calendar time — i.e., a task must be done every three months, regardless of when the task was last performed.
- Elapsed time — i.e., a task must be done within 60 days of

the last time it was done.

#### Time-based

Time and meter based scheduling require an input of machine running hours/kilometres. The PASS system can supply a number of the running hours values required for work order generation. The running hours from PASS are placed into a buffer, which can be edited by the maintenance engineer. If PASS data is not available, this buffer can be keyed in directly from manual meter readings.

- Fixed meter measurements — i.e., a task must be done every 700 hours of operation, regardless of when the task was last performed.
- Elapsed meter measurements — i.e., a task must be done within 500 hours of the last time it was done.

#### Consumption-based

Consumption-based measurements are similar to meters; however, they require a computation beyond simple tracking of operation. The most likely requirement would be work orders generated by fuel consumption or lubricating oil consumption. This requires the ability to compare expected fuel, oil, or chemical levels with actual tank soundings or inventory checks. The system has the ability to schedule tasks based on recorded consumption of fuels, lubricating oils, and chemicals versus standard consumption levels.

#### Condition-based

The use of condition monitoring techniques is well-established in the many industrial plants. Most programs involve one or more of lubricant analysis, vibration analysis,



performance monitoring, etc. The PUMP system has the ability to schedule work orders based on condition monitoring assessments produced by the VARMINT system. In figure 2, a list of diagnostics for a piece of equipment is displayed in PUMP's Equipment Manager, along with the underlying condition monitoring data that produced the diagnostic.

### Opportunity-based

Opportunity-based maintenance is scheduled by linking any task to one or more associated tasks. The activation of the first task can then trigger any or all of the associated tasks. This enables the maintenance engineer to take advantage of the under repair status of the equipment to get other work done on the same piece of equipment or on some other machine somehow related to the down machine.

The only risk involved in using opportunity based scheduling is a "circular reference", i.e.,

- task A is scheduled if task B is scheduled.
- task B is scheduled if task A is scheduled.

In the absence of any other scheduling rules, neither of these tasks will ever get scheduled! This is a situation that must be carefully avoided.

### Operation-based

Operation-based maintenance uses the values found in the systems' "observation sets." These are used to schedule maintenance based on *operational state* information. This information can be date-based, such as the ship's operational plan or the ship's staff change schedule, or it can be

situational, such as defining which set of inspection requirements are applied to this ship.

The task scheduling system has the flexibility to produce maintenance schedules based on a multitude of criteria. However, for a system with such a flexible structure, it is surprisingly simple to operate. The operator can choose from a pre-set list of rules when determining the schedule for a task. The system has a "look-forward" mode which allows the user to see how his scheduling for a particular task will take place over the months and years ahead—given some assumptions about the operation of the ship, of course. Finally, the system has full context sensitive help, allowing the operator to get assistance for the part of the program he/she is currently working with.

## 4. Work Orders, Reports and Queries

One of the main requirements of a maintenance management system is the ability to deliver work orders that the maintenance professional finds useful to complete the task in question. This means that work orders should have all of the necessary information needed to complete the task *and not one thing more*. This type of tightly focused work order can be difficult to achieve in many maintenance management systems where you are restricted to a single work order type—this necessitates an "everything plus the kitchen sink" approach to the work order form, so that complicated tasks can be defined correctly.

The PUMP system resolves this problem by producing both *primary* and *secondary* work orders forms. A primary

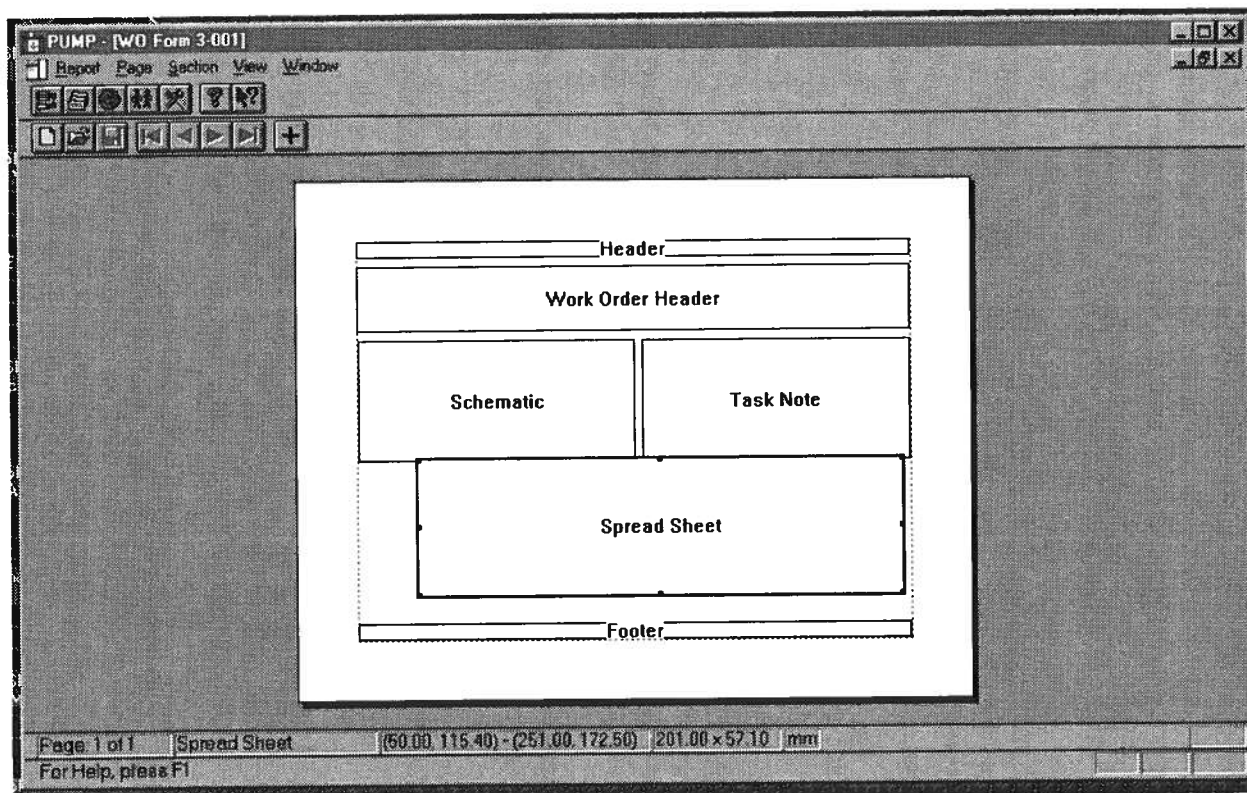


Figure 5. Work Order Report Editor

form is printed for all work orders generated by the system; secondary work orders are produced only for those work orders that require more information than is available on the primary form. Primary work orders are printed on 3"x5" card stock, and contain the minimum amount of information required to perform the job. Notes can be written on the back of the card. Secondary work orders are defined using the system work order editor, and the maintenance engineer can define many types of secondary work order forms as are required, including the generation of data entry forms and tick-lists. Figure 5 shows the Work Order Editor.

Detailed secondary work orders will allow the maintenance engineer to print (if required):

- task procedures
- estimated manpower requirements
- parts required
- tools required
- contractor work required
- all other charges
- detailed drawings of machinery
- scanned images of machinery
- associated system lockout requirements
- any forms required for special task classifications, such as environmental impact reports for tasks involving possible risk to the environment, or reports to the occupational safety authorities if the task has a safety aspect.

For major repair type work orders, tasks can be printed out in a "rows and columns" type format. Engineers should have the ability to fill in any pertinent data that can be easily entered into the system.

Another requirement for acceptance of a maintenance management system is ease of work order closing. The PUMP system is designed to allow the maintenance engineer to complete a work order with minimal effort. The main method used to close a work order is to scan a bar code on the primary work order. The maintenance engineer can then record actual usage and actual costs data for the work order. The system should accept the system standard work order parts and labour costs as actual costs unless overridden by the maintenance engineer.

Once a work order has been completed, the PUMP system has an easy-to-use interface to query work order history. The engineer who is looking for a history of the work performed on a piece of equipment can simply find the equipment record by searching a list of equipment, searching a hierarchy by location or system, entering a query to look for one of more equipment attributes, or by a "point and shoot" search through the ship's schematic. Once the equipment record has been selected, all work order history, including parts, personnel, and contractor rates and hours, is available by pressing one button. This simplifies the task of looking for work history, especially if you are unsure of the equipment code used on the ship.

PUMP has a full SQL-like query system, with output being produced either as a worksheet on the screen or as a table-like report. A report editor similar in function to the

work order editor is used to define both the layout and the content of the report. This is especially useful for crafting reports designed to meet the requirements of regulatory agencies, such as OSHA, insurance underwriter's, or environmental agencies. A flexible query/report system is useful, as it is difficult to determine what information will be required by these agencies over several years.


## 5. Benefits to the User

The vibration and lubricating oil analysis systems have already saved the CCG substantial amounts in maintenance and operating costs as stand-alone systems. It is expected that with time and operating experience, PUMP's ability to integrate data and produce improved advisory information, scheduling options, and flexible reports will pave the way for additional reductions in operating cost.

It is expected that maintenance personnel who are partly dedicated to parts management and inventory control will be freed to be deployed elsewhere, increasing the efficiency of the maintenance staff within the ship's crew.

An area that may lead to major cost reductions in the maintenance of CCG vessels is current practices carried out by safety inspectors. The current practice calls for a series of fixed survey periods for operating equipment, ranging from one year up to eight years, depending on the equipment in question. Classification societies such as ABS and Lloyd's have, over the last five years, begun to document the on-ship preventative and predictive maintenance methods that should be applied in order to be granted extensions to these survey periods.

Rotating and reciprocating equipment on CCG ships often have very low running hours in a survey period, relative to deep sea vessels. Yet the survey periods are applied regardless of hours of operation—they are based strictly on calendar time. Ships Safety Branch inspectors have, in the past, granted extensions to survey on machines based solely on low running hours and confidence in the shipboard maintenance staff. However, both SSB and CCG staff have expressed a desire for a more structured approach to survey extensions—and using predictive and preventative maintenance methods are seen as an effective tool to base survey period extension decisions on empirical evidence, rather than gut feel.

The PUMP system, in combination with performance analysis, vibration analysis, lubricant testing, and cylinder pressure monitoring, meets and greatly exceeds the specified requirements for a monitoring and maintenance tracking system as detailed by the class societies. The use of PUMP and its subordinate data analysis subsystems will go a long way in producing the type of maintenance data that can lead to a systematic approach to survey extension. 

*Steve Reilly, Design Maintenance Systems Inc., Vancouver, B.C.*

# Hybrid Intelligent Systems and Diagnostic Applications

*Sumit Virmani and Larry Medsker*

## Résumé

L'intégration de technologies intelligentes attire de plus en plus l'attention des chercheurs, des développeurs d'applications et des gestionnaires. Individuellement, les réseaux neuronaux, les systèmes experts, la logique floue, les algorithmes génétiques et le raisonnement basé sur des cas ont prouvé leur utilité dans le développement d'applications réelles. Cet article présente un survol de l'intégration de ces technologies intelligentes et passe en revue l'état et l'importance des applications de systèmes intelligents hybrides, avec une attention spéciale pour les systèmes de diagnostic. On y inclut une liste des ressources et des groupes de R&D actifs dans le domaine des systèmes hybrides.

## Abstract

The integration of intelligent technologies is steadily gaining attention of researchers, applications developers, and managers. Individually, neural networks, expert systems, fuzzy logic, genetic algorithms, and case-based reasoning have proven to be useful for developing real-world applications. This article presents an overview of the integration of these intelligent technologies and a review of the status and significance of hybrid intelligent systems applications, with an emphasis on diagnostic systems. An overview is included of the resources and R&D groups active in the hybrid systems area.

## Keywords

Intelligent systems, hybrid systems, expert systems, neural networks, fuzzy systems, genetic algorithms, case-based reasoning, intelligent information systems.

## 1 Overview of Intelligent Technologies

Several intelligent computing technologies are becoming useful as alternate approaches to conventional techniques or as components of integrated systems. This article is an overview of the status of and prospects for applications of hybrid systems. Further details can be found in [Goonatilake & Khebbal, 1995; Medsker, 1995; Medsker, 1996].

Hybrid intelligent systems are usually implemented using traditional computing systems. Powerful hybrid systems using expert systems and networks are well established [Medsker, 1995; Medsker, 1994]. Currently, developers are exploring and using these and other combinations of intelligent technologies, such as fuzzy logic, genetic algorithms, and case-based reasoning. Intelligent systems have a natural synergism that can be exploited to produce powerful components of computing systems. They may attempt to make up for deficiencies in the conventional

approaches or make better, more efficient, and effective computing systems. The appropriate use of intelligent technologies leads to useful systems with improved performance or other characteristics that cannot be achieved through other methods. The impact on organizations range from new products and manufacturing techniques to more realistic systems to support management decision making.

### 1.1 Expert Systems

Expert systems perform reasoning using previously-established rules for well-defined and narrow domains. Rule-based systems combine knowledge bases and domain-specific facts with information from clients or users about specific instances of problems in the knowledge domains of the expert systems. Ideally, the reasoning can be explained, and the knowledge base can be modified independently. This has led to the creation of expert system shells that keep the knowledge base separate from the details of the reasoning mechanism. Another feature of an expert system is the explanation facility, which allows the user to inquire about the reasoning process and the conclusions being presented by the expert system. For stable applications with well-defined rules, practical expert systems are readily produced and can provide excellent performance.

### 1.2 Neural Networks

The state of the art in neural computing is inspired by our current understanding of biological neural networks. The artificial neuron needs to evaluate its inputs by obtaining a weighted sum of the inputs from the simulated dendrites and to determine the level of the output on the simulated axon. The individual artificial neurons are combined into networks that are useful for computing.

Information processing with neural computers consists of analyzing patterns of activity using learned information stored as weights between node connections. A popular architecture is the multi-layered feedforward network. It has one layer to represent input data and another to represent its corresponding output. Between these layers, one or more intermediate or hidden layers containing a variable number of nodes can represent complicated, non-linear relationships between inputs and outputs. Commonly, each input node is connected to each node in the first hidden layer, and each node in the hidden layer is connected to every node in the following layer. Since a weight is associated with each connection, typical networks have a large matrix of weight values. A set of weights is adjusted in the training phase, using the back-error propagation algorithm or some variation of it.

Multi-layered networks using back-error propagation are prime examples of supervised learning. The other type is unsupervised learning, in which the network places the input vectors into categories without desired output vectors being supplied.

A characteristic of neural networks is the ability to classify streams of input data without the explicit knowledge of rules and to use arbitrary patterns of weights to represent the memory of categories. This gives neural networks the potential to provide some of the human characteristics of problem solving that are difficult to simulate using the logical, analytical techniques of expert system and standard software technologies.

### 1.3 Fuzzy Systems

Fuzzy systems are based on fuzzy set theory and associated techniques pioneered by Lotfi Zadeh [Munakata & Jani, 1994; Voit, 1993; Zadeh, 1965; Zadeh, 1994]. A goal of this approach is to mimic the aspect of human cognition that can be called approximate reasoning. Fuzzy systems may be less precise than conventional systems but are more like our everyday experiences as human decision makers. They can be used independently or in conjunction with other software such as hybrid intelligent systems, particularly fuzzy expert systems. In fuzzy expert systems, rules link fuzzy input and output variables. If the input is not fuzzy, a preprocessing module transforms the data to fuzzy values using membership functions. Then, the relevant fuzzy rules are applied and the actions with corresponding membership values are determined. Finally, the fuzzy output, possibly the results of some type of an averaging procedure, is transformed to a non-fuzzy value.

A number of applications of fuzzy logic have recently appeared in consumer products, initially in Asia and Europe and now in the U.S. [Walsh, 1993]. Examples include autofocus cameras, washing machines, and microwaves. Recently, the interest in applying fuzzy logic has extended to problems in information processing and managerial decision making. Also, special software packages for developing fuzzy systems are becoming available and should markedly increase the number of such systems in use.

### 1.4 Genetic Algorithms

This technology seeks to represent intelligent systems by mimicking the way biological systems self-organize and adapt to their environments [Goldberg, 1994; Hedburg, 1994; Lawton, 1992]. Genetic algorithm systems use feedback from the interaction with the environment to find adequate solutions to problems. Feedback causes better candidate solutions to survive as certain solutions to a problem drop out of consideration given weak or negative feedback.

Genetic algorithms can be considered as an alternative search technique. They can find locally optimal solutions that are adequate, thereby removing the need to do exhaustive searches. This is good for complex situations in which the search space would be prohibitively large for conventional

techniques.

Problems to be solved with genetic algorithms need to have solutions that can be described as a string of numbers or characters, where each symbol may represent more complex operations. This string is called a chromosome and has certain properties that give it a relative worth as a solution to a particular problem. An evaluator function produces a numerical value to represent the chromosome's ability to solve the problem. A generator function uses the crossover and mutation operators, which have both algorithmic and random aspects, to reconfigure the parents into new candidate solutions. The results of the evaluator are used to weed out solutions below a threshold and create a new pool of candidate solutions. Genetic algorithms have good potential for solving organizational problems such as scheduling and resource allocation [Lawton, 1992].

### 1.5 Case-Based Reasoning

This relatively new field started in the middle of the 1980's and is based on the idea of making use of solutions to previous problems for solving new ones. Development systems are now available to help analyze historical information and put it into a form that is useful for subsequent problem solving [Allen, 1994].

Case-based reasoning (CBR) was conceived as a technique similar to the aspect of human reasoning in which we refer to past experiences for guidance in solving current problems. CBR systems store solutions to previous problems and determine the differences from new problems. They analyze a problem and by means of an indexing system, retrieve matching stored cases, along with their solutions. If a retrieved case is not a close enough match, an attempt is made to modify it and present the solution for the problem at hand. If the case is still inadequate, human intervention may be required to establish a new case and store it in the case base for future use.

### 1.6 Hybrid Intelligent Systems

All of the intelligent technologies have been investigated, to varying degrees, as the ingredients for hybrid intelligent systems. Predominantly, a hybrid system uses two of the technologies; however, a large application may use many modules of each type. Most of the hybrid systems work has combined expert systems and neural networks, fuzzy logic and expert systems, or fuzzy logic and neural networks. Recently, considerable interest is evident in systems integrating genetic algorithms with neural networks or with fuzzy systems. Other combinations are at early stages of investigation and use.

As described in [Medsker & Bailey, 1992], the integration of intelligent systems can be viewed according to five models. The standalone approach uses each technology separately to study and design an application, and validate the design regardless of the final delivery system. The transformational model uses one technology in the design process with the intention of using the other for final implementation. In

some cases, such as expert networks, the second system (neural network) is transformed back to the initial technology (expert system) as the delivery system [Kuncicky, Hruska, & Lacher, 1992]. Loose and tight coupling models have delivery systems with distinct modules that use intelligent technologies, with loose coupling using file transfer and tight coupling using internal data structures and function calls to pass data between the modules. Fully integrated systems merge the intelligent technologies resulting in a new type of system without separate intelligent modules.

## 2 Hybrid Intelligent System Applications for Diagnosis and Repair

### 2.1 Expert Systems and Neural Networks

The most well-established area of hybrid systems is the integration of expert systems and neural networks. Many applications have been developed in the last five years, and some examples relevant to the focus of this article are listed below.

#### Condition Monitoring System for Power Plants

The use of hybrid neural network and expert systems in the power industry is described in [MacIntyre, Smith, and Harris, 1995]. They focus on the application of such systems on the problem of monitoring the health of machines. Condition monitoring involves the capture of data that describes various parameters of machinery and the analysis of this data to determine the condition of a given machine at a certain time. The main parameters to be monitored include vibration in rotating machinery, temperature in mechanical systems, and the analysis of lubricating oil and grease. As shown in Figure 1, a neural network component can be used to process these types of data for further use in an expert system component.

#### Nuclear Power Plant Monitor

A tightly-coupled hybrid system of neural networks and expert systems was developed for a sensor-monitoring system to support nuclear power plant operators [Mazzu, Caglayan, and Gonsalves, 1994]. The system was created using a Macintosh-based NueX hybrid system development environment. It shows good potential for enhancing operator efficiency and performance, improving plant operations by early detection of off-normal states, and improving operator training.

Neural networks are employed to detect and isolate flux detector failures, which consist of subtle temporal and spatial changes. Knowledge-based systems are used to determine more drastic detector failures, interpret the neural network results, and provide overall monitoring assessment.

The state detection neural network analyzes weekly flux measurements and an expert system interprets the results. A database of these results is then analyzed by another expert system, taking into account correlated detectors. Concurrently, another knowledge-based system looks at the neutron flux data for evidence of hard failure states. The last expert system makes the final determination of each detector's operating state, presents the results through the graphical user interface, and compares the assessment with the detector's known state for system evaluation purposes.

#### Expert Networks — The Control Chart Selection Advisor

One example of the use of expert networks is called the Advisor for Selection of Control Charts (ASCC) [Hruska and Whitfield, 1995]. The ASCC is an expert network designed to capture expert advice regarding eleven control charts and their appropriate use for tracking the state of a particular manufacturing process.

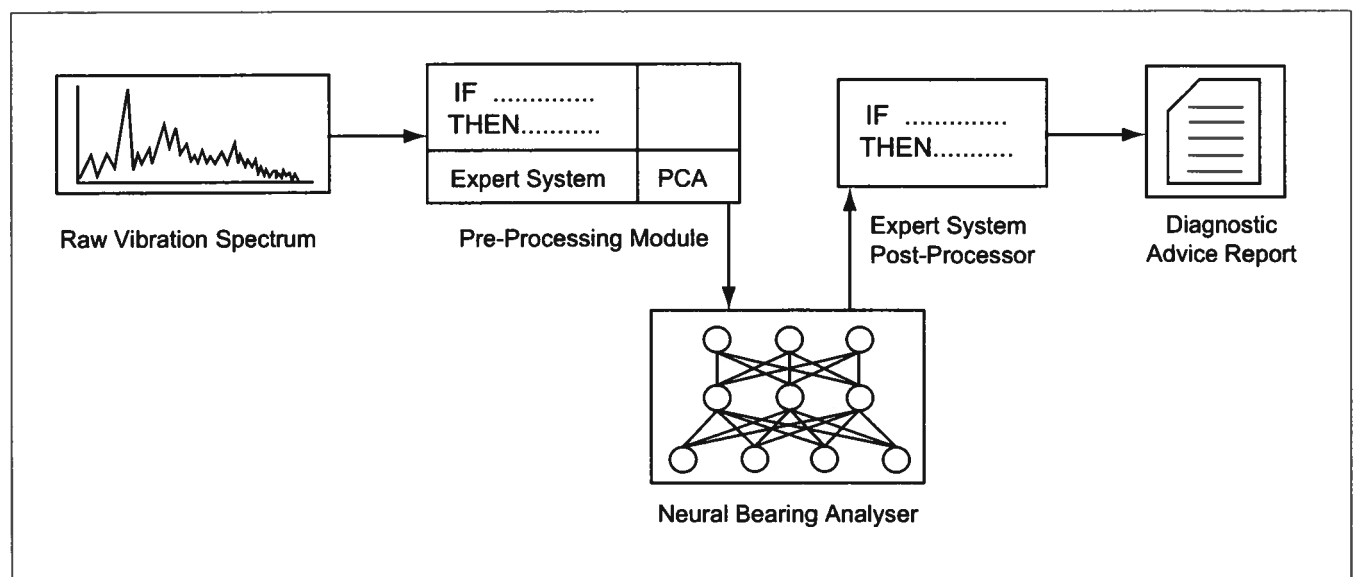


Figure 1. The hybrid neural network and expert system for monitoring vibration data.



The basic concept behind the function of ASCC is that decisions are made based on initial input acquired from thirteen questions reflecting the setup of a manufacturing process, the type of data and measurements being gathered, and the available resources of the engineer. As a result of the initial input, ASCC recommends the appropriate control chart by prioritizing the set of eleven charts based on certainty factors. Expert network training can make use of example data to take over the task of setting certainty factors at which human experts do not excel, especially when there is a relatively complex set of rules to be considered.

The theory of expert networks blends the familiarity of rule-based expert systems with the power of neural network style training. The resulting applications show that this hybrid system is effective in attacking many of the common problems of both expert systems and neural networks, and can be used to build effective knowledge-based systems. Capitalizing on the strong points of more than one type of knowledge-based system makes for robust applications which are adaptable to data. Hybrid systems free their human creators from the need to mold their problem to a single type of solution, and allow them to think more creatively about both problems and solutions [Hruska and Whitfield, 1995].

## 2.2 Fuzzy Systems and Neural Networks

Two important ways to combine fuzzy logic and neural networks is (a) the use of cooperating modules and (b) the implementation of fuzzy logic as a neural network. The latter is illustrated in Figure 2. As indicated in the figure, the different layers of the neural network represent different aspects of the fuzzy system. Training the neural network then optimizes the fuzzy system.

### Application of Neural Networks and Fuzzy Systems to Power Plants

Several examples of coupled systems are found in the work by Uhrig et al. [Uhrig, Tsoukalas, and Ikononopoulos, 1994]. They have investigated a number of applications of fuzzy, neural, and hybrid systems for improving the performance and safety of nuclear power plants. They see potential for hybrid systems that couple neural networks with fuzzy rule-based systems. The neural networks, which represent membership functions, map physical states to degrees of membership that are then used in fuzzy systems. Examples of promising applications using this approach to hybrid fuzzy and neural systems include identification of transient noise and the use in inferential sensors.

### Neural Network-Fuzzy Logic Diagnosis System for Vibration Monitoring

In [Loskiewicz-Buczak and Uhrig, 1993], the diagnosis of faults in roller bearings is based on the relative magnitude of the peaks occurring in spectra at characteristic frequencies (and their harmonics) associated with the bearing geometry

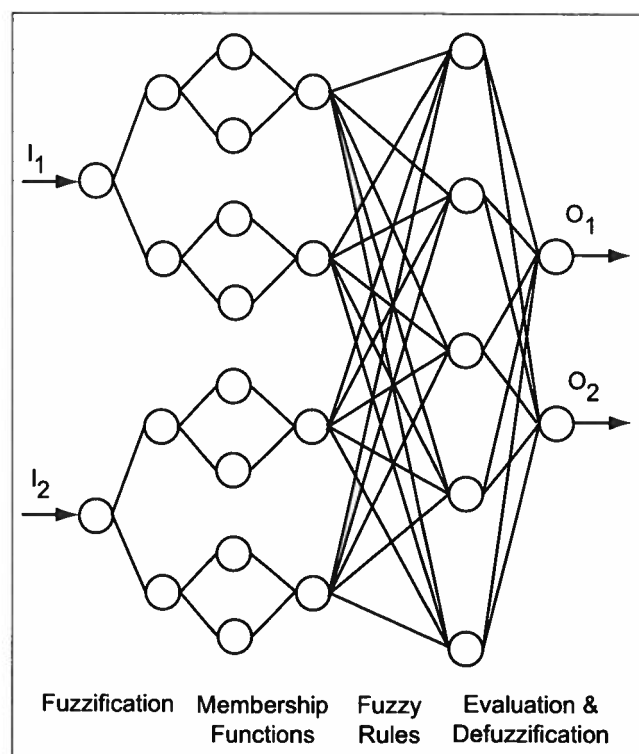


Figure 2. Architecture of a fuzzy neural network

and the basic rotating frequency. A comparison between the results using crisp magnitudes of the amplitudes and two types of fuzzy representation of these amplitudes is presented.

### Decision Fusion by Fuzzy Set Operations

In [Loskiewicz-Buczak and Uhrig, 1994], a composite diagnosis of single and multiple faults in rotating machinery is obtained based on the fusion of nine tentative diagnoses indicated by neural network processing of data from nine sensors placed on the individual machine. The fusion of these decisions is performed by a fuzzy logic connective called the Generalized Mean. The fusion process proceeds step by step using a "confidence factor" to determine whether or not to include each successive tentative diagnosis.

### Surface Failure Detection for an F/A-18 Aircraft

The work [Raza, Ioannou, and Youssef, 1994] on detecting surface failures for aircraft compares standalone fuzzy and neural systems. Systems using fuzzy controllers have been applied to oil refinery problems, power system stabilizers, and robot manipulators.

## 2.3 Genetic Algorithms and Fuzzy Logic

Genetic algorithms can be used to optimize the parameters or structure of fuzzy systems. As shown in Figure 3, populations of fuzzy system design specifications are implemented and tested against desired performance levels. The results are used to select and modify subsequent populations until a good fuzzy system is obtained.

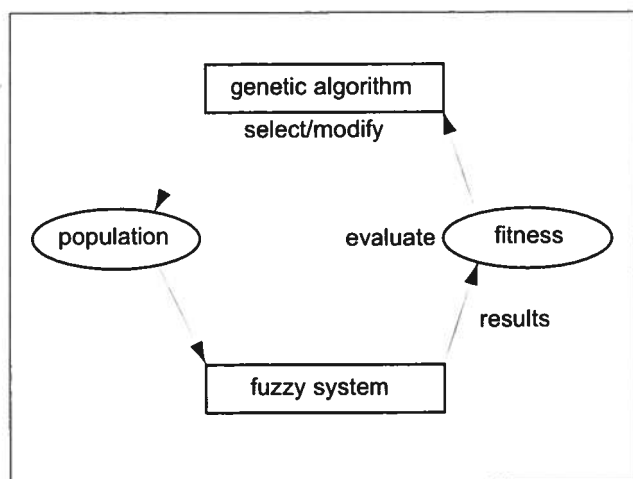


Figure 3. The process of using a genetic algorithm to improve a fuzzy system design.

### Uncertainty Management in Space Station Autonomous Research

Pal, 1993, conducts research to support space station autonomous operations for unmanned missions. In particular, he addresses the representation and management of uncertainties involved in image analysis and recognition tasks using various tools based on probability, fuzzy logic, and genetic algorithms. The approach shows potential in the area of soft decisions and the efficient handling of uncertainties in autonomous operations.

### Fuzzy Controller for a Plant

Barczak, Martin, and Krambeck, 1994, describe a fuzzy controller, based on a rule set for the strategy to control a plant, and a genetic-based rule-tuning process. With this approach, little has to be known about the plant to be controlled or about the best set of rules. For example, in servo-systems used for positioning machine tools, they find that a final set of rules can be the combination of straightforward rules plus those determined by the genetic algorithm.

## 2.4 Genetic Algorithms and Neural Networks

Several different aspects of the design of neural networks can be automated and improved using genetic algorithms. Among those are the selection of neural network input data, architectural choices such as number of layers and nodes, and initial values of parameters. Figure 4 shows the steps involved.

### Use of Genetic Algorithms to Select Inputs for Neural Networks

The work of [Guo and Uhrig, 1992] shows a use of genetic algorithms to improve the application of neural networks for fault diagnostics in nuclear power plant operations. In this work, genetic algorithms are used to

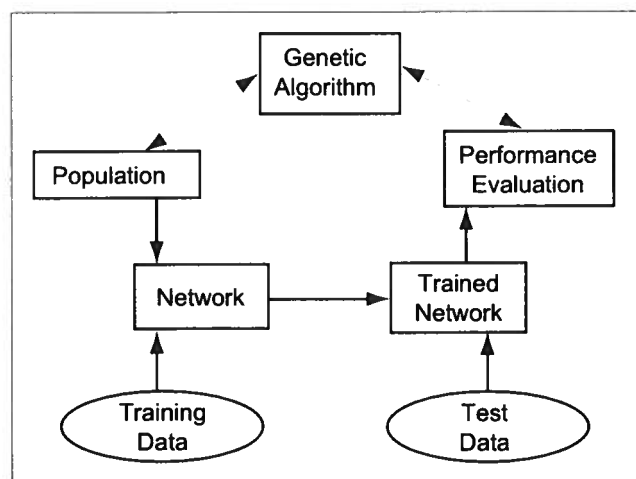


Figure 4. The process of using a genetic algorithm to optimize a neural network.

guide the search for optimal combination of inputs, from a large number of plant variables, for the neural networks to achieve faster training and improved accuracy with a minimized number of inputs. Data from the Tennessee Valley Authority WattsBar Nuclear Power Plant simulator are used to demonstrate this hybrid approach.

Loskiewicz-Buczak and Uhrig, 1993, have designed a diagnostic system that aggregates evidence from many sensors in order to automate the interpretation of machine vibration spectra. Their active decision system, which fuses decisions only from the sensors needed in a given situation, uses a fuzzy system module to perform the generalized mean operation, and the parameters are chosen by a genetic algorithm. The final decision is obtained from the aggregated decision by means of alpha-cuts. This is a general method that could be used for any classification problems involving the aggregation of decisions.

## 2.5 Case-Based Reasoning

Vo and Macchion, 1993, use case-based reasoning to understand knowledge domains for expert systems to support satellite development. One application is for the support of test engineers in the assembly, test, and validation phase. Another is for the diagnosis of problems in a vehicle equipment bay. One use of the hybrid system is to suggest causes of anomalies when an incident appears, based on characteristics relevant in similar past incidents. The system can also use rules and models in the proposal of a diagnosis approach. Karamouzis and Feyock, 1993, study the use of a hybrid case-based approach for in-flight fault diagnosis and prognosis for aviation subsystems. The system includes in the case-based reasoning a model of the physical system to describe the system's structural, functional, and causal behavior. [Tsutsui, Kurosaki, Sato, and Hiraide] have developed a fault detection system for chiller performance deterioration.

Description	Location	Electronic Address
Berkely Initiative in Soft Computing Dr. Lotfi A. Zadeh	University of California, Berkely	<a href="http://http.cs.berkeley.edu/projects/Bisc/bisc.welcome.html">http://http.cs.berkeley.edu/projects/Bisc/bisc.welcome.html</a>
AI Research Group Dr. Vasant Honavar	Iowa State University	<a href="http://www.cs.iastate.edu/~honavar/aigroup.html">http://www.cs.iastate.edu/~honavar/aigroup.html</a>
Intelligent Hybrid Systems Dr. Sukhdev Khebbal and Dr. Suran Goonatilake	University College, London	<a href="http://boom.cs.ucla.uk/staff/skhebbal/ihs">http://boom.cs.ucla.uk/staff/skhebbal/ihs</a>
Dr. C. Lee Giles	NEC Research Institute	<a href="http://www.neci.nj.nec.com/homepages/giles.html">http://www.neci.nj.nec.com/homepages/giles.html</a>
Special Interest Group on Hybrid Intelligence (SIGHI) Dr. Larry Medsker and Dr. Ron Sun	International Neural Networks Society	<a href="mailto:listproc@cs.ua.edu">listproc@cs.ua.edu</a>
Hybrid and Integrated Systems SIG Dr. Hossein Soltan	Cranfield University, England	<a href="http://www.cranfield.ac.uk/public/im/cm013/AI_CENTRE/HYIS/hyis.html">http://www.cranfield.ac.uk/public/im/cm013/AI_CENTRE/HYIS/hyis.html</a>
Hybrid Intelligent Systems Group Dr. Larry Medsker	The American University Washington, D.C.	<a href="http://www.cas.american.edu/research/his.html">http://www.cas.american.edu/research/his.html</a>
Machine Intelligence Institute Dr. Ron Yager	Iona College, New Rochelle, NY	<a href="http://www.iona.edu/mii.htm">http://www.iona.edu/mii.htm</a>
Integrated Reasoning Group Mike Halasz	Institute for Information Technology National Research Council, Canada	<a href="http://ai.iit.nrc.ca/IR_public">http://ai.iit.nrc.ca/IR_public</a>
Dr. Ron Sun	University of Alabama	<a href="http://www.cs.ua.edu/faculty/sun/sun.html">http://www.cs.ua.edu/faculty/sun/sun.html</a>


Table 1. Research and Development Groups

### 3 R&D Groups

Table 1 provides a list of research and development groups in the field of hybrid intelligent systems with their locations and electronic addresses.

#### References

- Allen, B. P. (1994), "Case-based reasoning: business applications," *Communications of the ACM* 37, pp. 40-42.
- Barczak, C. L., C. A. Martin, and C. P. Krambeck (1994), "Experiments in fuzzy control using genetic algorithms," *Proceedings of IEEE International Symposium on Industrial Electronics*, Santiago, Chile, May 25-27, pp. 426-428.
- Goldberg, D. E. (1994), "Genetic and evolutionary algorithms come of age," *Communications of the ACM*, vol 37, no. 3, pp. 113-119.
- Goonatilake, S., and S. Khebbal (1995), *Intelligent Hybrid Systems*, New York: John Wiley and Sons.
- Guo, Z., and R. E. Uhrig (1992), "Use of genetic algorithms to select inputs for neural networks," *Proceedings of the Special Workshop on Combinations of Genetic Algorithms and Neural Networks*, International Joint Conference on Neural Networks, Baltimore, MD, June 6-11.
- Harp, S. A., and T. Samad (1991), "Optimizing neural networks with genetic algorithms," *Proceedings of the American Power Conference*, Chicago, pp. 1138-1143.
- Hedberg, S. (1994), "Emerging genetic algorithms," *AI Expert*, vol 9, no. 9, pp. 25-29.
- Hruska, S. I., and D. C. Kuncicky (1992), "Automated Knowledge Refinement for Control Chart Selection," vol 5, *Heuristics*, pp. 101-107.
- Hruska, S. I., and Whitfield, T. A. (1995), "Expert networks: theory and applications," in Medsker, L., *Hybrid Intelligent Systems*, Boston: Kluwer Academic Publishers.
- Hung, C. C. (1993), "Building a neuro-fuzzy learning control system," *AI Expert* 8, pp. 40-49.
- Kadaba, N., K. E. Nygard, and P. J. Juell (1991), "Integration of adaptive machine learning and knowledge-based systems for routing and scheduling applications," *Expert Systems with Applications* 2, pp. 15-27.
- Kandel, A. (1992), *Fuzzy Expert Systems*, Boca Raton, FL: CRC Press.
- Kandel, A., and G. Langholz (eds) (1992), *Hybrid Architectures for Intelligent Systems*, Boca Raton: CRC Press.

- Kandel, A., and G. Langholz (eds.) (1994), *Fuzzy Control Systems*, Boca Raton, FL: CRC Press.
- Karamouzis, S. T., and S. Feyock (1992), "An integration of case-based and model-based reasoning and its application to physical system faults," *Proceedings of the 5th International Conference, IEA/AIE-92, Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, Paderborn, Germany, June 9-12, pp. 100-108.
- Karr, C. (1991), "Applying genetics to fuzzy logic," *AI Expert* 6, pp. 38-43.
- Kosko, B. (1992), *Neural Networks and Fuzzy Systems: A Dynamical Approach to Machine Intelligence*, Englewood Cliffs, NJ: Prentice Hall.
- Kuncicky, D. C., S. I. Hruska, and R. C. Lacher (1991), "The equivalence of rule-based expert system and artificial neural network inference," *International Journal of Expert Systems*, vol 4, no. 1, pp. 281-297.
- Lawton, G. (1992), "Genetic algorithms for scheduling optimization," *AI Expert* 7, pp. 23-27.
- Loskiewicz-Buczak, A., and R.E. Uhrig (1993), "Aggregation of evidence by fuzzy set operations for vibration monitoring," *Proceedings of the Third International Conference on Industrial Fuzzy Control and Intelligent Systems*, Houston, TX, December 1-3, pp. 204-209.
- Loskiewicz-Buczak, A. and R. E. Uhrig (1993), "Neural network-fuzzy logic diagnosis systems for vibration monitoring," *Proceedings of ANNIE-93, Artificial Neural Networks in Engineering Conference*, St. Louis, MO, November 14-17.
- Loskiewicz-Buczak, A., and R. E. Uhrig (1994), "Decision fusion by fuzzy set operation," *Proceedings of IEEE World Congress on Computational Intelligence, Fuzzy Systems Conference*, Orlando, FL, June 26-30, pp. 1412-1417.
- MacIntyre, J., Smith, P., and Harris, T. (1995), "Industrial experience: the use of hybrid systems in the power industry," in Medsker, L., *Hybrid Intelligent Systems*, Boston: Kluwer Academic Publishers.
- Mazzu, J. M., P. G. Gonsalves, and A. K. Caglayan (1994), "Hybrid system approach to nuclear plant monitoring," in [Medsker, 1994a] at pp. 77-108.
- Medsker, L. R.(ed.) (1991), *Special Issue of Expert Systems with Applications: An International Journal* 2. Medsker, L. R. (1994), "Design and development of hybrid neural network and expert systems," *Proceedings of the IEEE International Conference on Neural Networks III*, Orlando, FL, pp. 1470-1474.
- Medsker, L. R. (1995), *Hybrid Intelligent Systems*, Boston: Kluwer Academic Publishers.
- Medsker, L. R. (1994), *Hybrid Neural Network and Expert Systems*, Boston: Kluwer Academic Publishers.
- Medsker, L. R., and Bailey, D. L. (1992), "Models and guidelines for integrating expert systems and neural networks" in [Kandel and Langholz, 1992] at pp. 154-171.
- Munakata, T. and Y. Jani (1994), "Fuzzy systems: an overview," *Communications of the ACM* 37, pp. 69-76.
- Nauck, D. and R. Kruse (1994), "NEFCON-I: An X-Window based simulator for neural fuzzy controllers," *Proceedings of the IEEE International Conference on Neural Networks III*, Orlando, FL, pp. 1638-1643.
- Oliver, J. (1994), "Finding decision rules with genetic algorithms," *AI Expert* 9, pp. 33-39.
- Pal, S. K. (1993), "Uncertainty management in space station autonomous research: pattern recognition perspective," *Information Sciences*, vol 72, no. 1-2, pp. 1-63.
- Raza, H., P. Ioannou, and H. M. Youssef (1994), "Surface failure detection for an F/A-18 aircraft using neural networks and fuzzy logic," *Proceedings of the IEEE International Conference on Neural Networks*, vol V, *IEEE World Congress on Computational Intelligence*, Orlando, FL, June 28-July 2, pp. 3363-3368.
- Soucek, B., and the IRIS Group (eds) (1991), *Neural and Intelligent Systems Integration*, New York: John Wiley and Sons.
- Sun, R. (1994), *Integrating Rules and Connectionism for Robust Common Sense Reasoning*, New York: John Wiley and Sons.
- Sun, R. and L. Bookman (eds) (1994), *Computational Architectures Integrating Neural and Symbolic Processes*, Boston: Kluwer Academic Publishers.
- Tsutsui, H., A. Kurosaki, T. Sato, and Y. Hiraide (1994), "Fault detection using topological case based modeling and its application to chiller performance deterioration," *Proceedings of IEEE Instrumentation and Measurement Technology Conference - IMTC '94*, vol 1, Hamamatsu, Japan, May 10-12, pp. 390-393.
- Uhrig, R. E., L. H. Tsoukalas, and A. Ikononopoulos (1994), "Application of neural networks and fuzzy systems to power plants," *Proceedings of the IEEE International Conference on Neural Networks*, vol VI, *IEEE World Congress on Computational Intelligence*, Orlando, FL, June 28-July 2, pp. 3703-3718.
- Vo, D. P., and D. Macchion (1993), "A use of case-based reasoning technique in building expert systems," *Future Generation Computer Systems*, vol 9, no. 4, pp. 311-319.
- Walsh, B. (1993), "Fuzzy logic: the American market," *AI Expert*, vol 7, no. 11, pp. 35-39.
- Wilke, P. (1993), "Simulation of neural networks and genetic algorithms in a distributed computing environment using NeuroGraph," *Proceedings of the World Congress on Neural Networks I*, Portland, OR, pp. 269-272.
- Zadeh, L. A. (1965), "Fuzzy sets," *Information and Control* 8, pp. 338-353.
- Zadeh, L. A. (1994), "Fuzzy logic, neural networks and soft computing," *Communications of the ACM* 37, pp. 77-84.
- Zurad, J. M., R. J. II Marks, and C. J. Robinson (eds) (1994), *Computational Intelligence: Imitating Life*, New York: IEEE Press. 

**Larry Medsker** is a professor of Computer Science and Information Systems at American University. His research and teaching focus on expert systems, neural networks, and the integration of intelligent technologies. He has published three books and more than 100 scientific papers. He earned a PhD in Physics from Indiana University and has held positions at the Purdue School of Science, University of Pennsylvania, and Bell Laboratories.

**Sumit Virmani** works with United Information Systems, Inc. (UIS) in Bethesda, MD as an Oracle Systems Development Specialist. His work with UIS focuses on building state-of-the-art information management and dissemination applications to help the National Cancer Institute meet their objective of world-wide availability of cancer information. At American University, his research involved intelligent system technologies including expert systems, neural networks, fuzzy logic, genetic algorithms, and case-based reasoning. He has developed hybrid and intelligent system applications using EXSYS Pro, NeuralWorks Pro, NeuroShell2, and LEVEL 5.

# A Fault Diagnosis Knowledge-Based System For Manufacturing Equipment in a CIM Environment

Aldo Dagnino and Brian Schack

## Résumé

Un système à base de connaissances pour le diagnostic des pannes d'équipement manufacturier dans un environnement FIO (fabrication intégré par ordinateur).

On présente dans cet article un système de diagnostic des pannes d'équipement manufacturier dans une usine de fabrication intégrée par ordinateur (FIO). L'objectif de ce système est de prêter assistance à l'opérateur humain pour déterminer les causes des pannes de machines. Le système, résultat d'un projet avec une compagnie multinationale de fabrication en télécommunication, a été développé pour dépanner une machine d'insertion Universal Multi-Mod. Cette machine assemble des composantes électroniques à deux rangées de pattes sur des plaques de circuits imprimés. Le MFADES (système expert de diagnostic de pannes de fabrication) comporte un module heuristique de diagnostic avec des capacités de raisonnement approximatif et un module de diagnostic causal profond.

## Abstract

A knowledge-based system for fault diagnosis of manufacturing equipment in a computer-integrated manufacturing (CIM) factory is presented. The objective of this system is to provide assistance to a human operator to determine the causes of machine failures. The system has been developed to trouble-shoot a Universal Multi-Mod insertion machine and is the product of a project with a multinational telecommunications manufacturing company in Canada. The Universal Multi-Mod insertion machine assembles electronic components with two rows of pins in printed circuit boards. The Manufacturing FAult Diagnosis Expert System (MFADES) contains a heuristic diagnostic module with approximate reasoning capabilities and a deep/causal diagnostic module.

## Introduction

This paper presents the structure of a knowledge-based diagnostic and preventive maintenance system which will be employed to maintain manufacturing equipment in a CIM factory. The system incorporates a heuristic knowledge module, a deep/causal knowledge module and different types of reasoning mechanisms which drive the inference engine. The first phase of the system has been developed based on a project developed by the Alberta Research Council (ARC) for a multinational telecommunications manufacturing company in Canada. The scope of this project was to develop a heuristic diagnostic system (MFADES), capable of assisting an operator in troubleshooting a Universal Multi-Mod insertion machine. This machine is common throughout the

corporation and it has a complicated set of troubleshooting procedures. The objective of the second phase of the project, is to enhance the MFADES system and develop a generic fault diagnosis architecture to troubleshoot and maintain several types of manufacturing equipment in a CIM factory. The new system incorporates a heuristic knowledge module, a deep/causal module and a maintenance module. This paper outlines the enhanced version of MFADES and shows an example of the results obtained in the development of the heuristic module for the Multi-Mod machine. Several of the concepts presented in this paper are derived from the work of Torasso and Console (1989). Presently, the heuristic level of the system has been implemented in MFADES employing the expert system shell Kappa-PC from Intellicorp.

## Classification of Diagnostic Knowledge-Based Systems

Diagnostic knowledge systems have been developed based on different types of knowledge representation. These types of knowledge systems include heuristic classification systems, causal and deep systems, and hybrid diagnostic systems.

### Heuristic Classification Systems

These systems contain "experience" or heuristic knowledge about the relationship between the findings of the system analyzed and its possible diagnoses. Three different phases can be recognized in the process of heuristic classification. In the *data abstraction phase*, data are collected from the environment and are employed by the diagnostic system. In the *heuristic match phase*, the system matches the data abstracted from the environment to the hierarchical classification of the faults. This matching is performed by employing control mechanisms that contain a variety of reasoning (forward, backward chaining, approximate reasoning, etc.) and knowledge representation techniques (production rules, frames, hybrid, etc.). The *refinement phase* refers to achieving a more accurate interpretation of the diagnosis. It consists of reasoning about the faults classification hierarchy to obtain a more refined solution to the diagnostic problem.

### Causal and Deep Systems

The structure and behaviour of the system being diagnosed are explicitly represented in causal and deep diagnostic systems. "Causal/temporal relationships can be effectively used to describe the behaviour of the modelled system in a

very precise way" [Torasso and Console, 1989]. There are two approaches to represent a system. The first approach describes the correct behaviour of the system and its components. The second approach describes the faulty behaviour of the system.

### Hybrid Diagnostic Systems

In a hybrid system, both heuristic and causal/deep knowledge are present. During the problem -solving process, first these systems employ heuristic knowledge to narrow the solution space and then employ causal/deep knowledge to refine the solution. The coordination and effective use of the different types of knowledge is the major challenge of hybrid systems. Nevertheless, these systems provide a more flexible and effective way of diagnosis.

### Architecture of MFADES

The main objective of the work developed in this project consists of building a knowledge system architecture capable of performing diagnostics and preventive maintenance of manufacturing equipment in a CIM environment. The system's architecture must be robust and flexible enough to accommodate different types of manufacturing equipment. This is achieved by incorporating to the basic architecture of the system, the required knowledge modules relative to each piece of manufacturing equipment. In other words, given any changes in the manufacturing equipment to be diagnosed, the architecture of the diagnostic system would not require modification. The only changes would be made to the knowledge bases that contain knowledge about the particular manufacturing equipment to be diagnosed.

Figure 1 shows the architecture of the diagnostic and maintenance system proposed in this work. The system has been designed to perform diagnostics of equipment at two different levels: heuristic level and deep/causal level. To perform these two types of diagnostics, a heuristic knowledge module and a deep/causal knowledge module are required in the system. The heuristic knowledge module contains heuristic knowledge gained by experts when they maintain and diagnose the manufacturing equipment of the plant. This heuristic knowledge associates diagnostic hypotheses with findings or conditions of the piece of equipment to be diagnosed. The deep/causal knowledge module consists of connected frames that explain the behaviours and connections of the components of the manufacturing equipment being diagnosed. The deep/causal level employs the initial set of hypotheses identified at the heuristic level and confirms or rejects them, by finding all the connections in the deep/causal level.

### Heuristic Knowledge Module

The heuristic knowledge module contains five sub-modules as shown in Figure 1: (a) heuristic hypotheses activation rules; (b) heuristic hypotheses frames; (c) heuristic hypotheses validation rules; (d) preventive maintenance rules; and (e) approximate reasoning mechanism and inference engine.

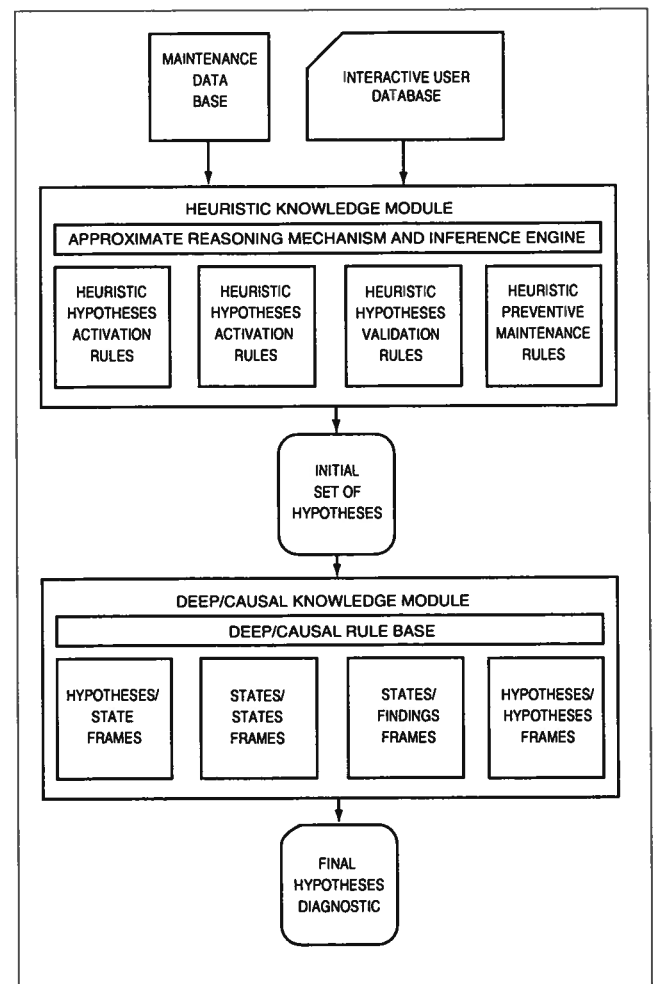


Figure 1. Architecture of Fault Diagnosis and Maintenance Knowledge System for Manufacturing

The inputs for this level of diagnostics are given in an interactive user interface, where the analyst provides information about a case to be diagnosed. For preventive maintenance analyses, a large database that contains information about equipment work load, is employed as input for the system. The output provided at the heuristic level of diagnostics is a set of plausible hypotheses associated with an equipment failure. Torasso and Console [1989] do not define a preventive maintenance module, which is incorporated in the system proposed in this work. The heuristic knowledge level modules are explained below.

### Heuristic Hypotheses Activation Rules

The heuristic hypothesis activation rules sub-module together with the heuristic validation rules module represent the control knowledge of the heuristic knowledge level of the diagnostic system. The heuristic hypotheses activation rules module contains a hierarchical set of rules that open a hypothesis frame if they are satisfied. The rules are given in a tree structure and map the diagnostic hypotheses considered by the system. At the higher level of the tree coarse diagnostic hypotheses are represented while at the lower level of the



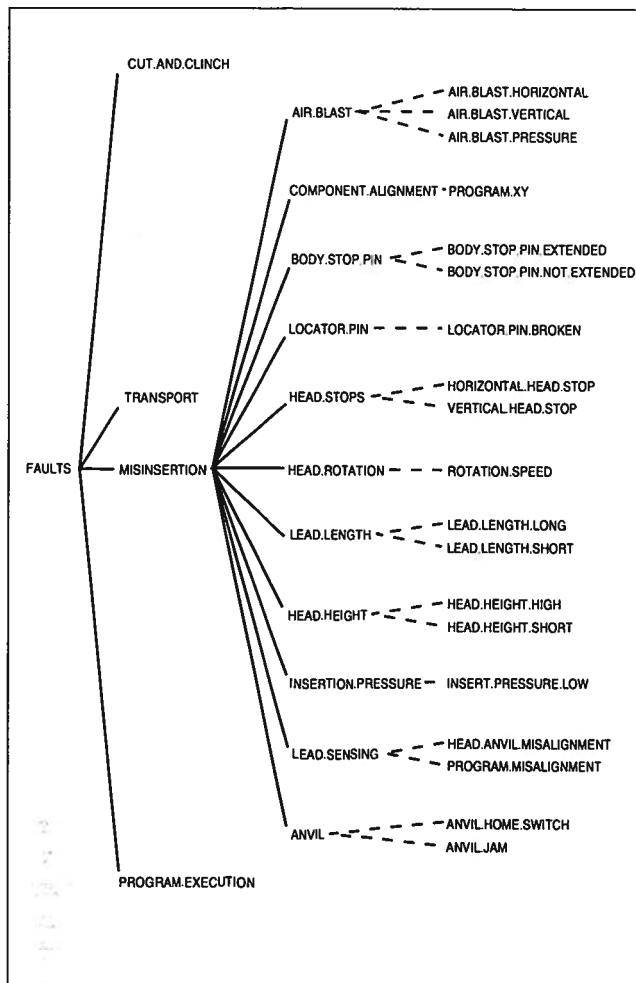


Figure 2. Universal Multi-Mod Insertion Machine Failure Tree Structure

tree more refined diagnostic hypotheses are connected. Associated with each activation rule a relevance measure is given. A relevance measure is defined in the interval [0.1], and allows the development of rules at different levels of specificity, which are then considered differently when the activation rules are evaluated (Torasso & Console, 1989).

### Heuristic Hypothesis Frames

This sub-module contains the frames associated with the diagnostic hypotheses considered by the system [Torasso and Console, 1989]. The heuristic hypotheses frames module contains the prototypical knowledge that describes the set of attributes that characterize a finding. The structure of this module is analogous to the heuristic hypotheses activation rules. Hypotheses frames are connected in a tree structure where coarse diagnostic hypotheses are defined at the higher level of the tree, and more refined diagnostic hypotheses are defined at the lower level of the tree. A heuristic hypothesis frame contains four types of slots: (a) necessary conditions; (b) supplementary conditions; (c) alternative hypotheses; and (d) specializations.

Necessary conditions slots represent the necessary

conditions or situations which must be present to confirm the diagnostic hypothesis activated. Necessary conditions are "ANDed." Supplementary conditions describe the situations that support the necessary conditions for the hypothesis frame activated. Supplementary conditions are "ORed." Torasso and Console [1989] explain the role of Sufficient conditions as well. Alternative hypotheses is a slot that suggests to the system other hypotheses similar to the one under consideration. This slot is accessed whenever the hypothesis activated fails. The specialization slot contains the specialization diagnostic hypotheses below the one under consideration. This slot is triggered whenever the hypothesis under consideration has been successful. It allows the system to continue further exploration towards more refined hypotheses.

MFADES' heuristic hypotheses leaf node frames contain two more slots: DIAGNOSIS slot and CORRECTIVE.ACTION slot. The function of these two slots is to provide a more accurate diagnostic and its corrective action to the operator of a Multi-Mod insertion machine.

### Heuristic Hypotheses Validation Rules

As mentioned earlier, this module is part of the control knowledge for the heuristic knowledge level. Validation rules can either confirm or exclude a hypothesis. Confirmation rules and rejection rules are mutually exclusive. Validation rules have a *relevance measure* associated that provide a measure for the strength of confirmation or exclusion of the hypothesis [Torasso & Console, 1989].

### Heuristic Preventive Maintenance Rules

This module contains a set of rules and a set of frames that have descriptive knowledge about the preventive maintenance operations for particular pieces of equipment. This module employs a large database as an input that contains information about volume of work processed by the various pieces of equipment, and past, present, and projected work loads of manufacturing equipment. Information about the past failure rates, costs, and down time is included in this module. The function of this module is to analyze the information contained in the database and to advise the maintenance staff about preventive maintenance of pieces of equipment.

### Approximate Reasoning Mechanism and Inference Engine

This sub-module contains the inference mechanism that manipulates the knowledge contained in the heuristic knowledge module. More details about this mechanism are given below.

### Deep/Causal Knowledge Level

The deep/causal knowledge level can describe the "correct behaviour" of a machine or the "faulty behaviour" of a machine. The model represented in the deep/causal level of the system proposed describes the faulty behaviour of a

machine or piece of manufacturing equipment. The reason behind this approach is that the manufacturing equipment to be diagnosed is very complex and many variables such as physical, chemical, electronic, mechanical, etc. must be considered when defining a correct behaviour model of the machines. Hence, the deep/causal knowledge module is designed to describe the malfunctioning of the machine being diagnosed.

In Torasso and Console [1989], the deep/causal module is defined as a network. The deep/causal module in the system proposed in this work is defined as a frame structure. Four different types of frames are defined. These frames include the following: (i) hypotheses frames; (ii) states frames; (iii) initial causes frames; (iv) findings frames.

Hypotheses frames in the deep/causal level are connected to the heuristic hypotheses frames. States frames represent the situation of a system at a particular point in time. Initial causes frames represent the possible initial causes of a disorder in a system. Findings frames represent observable conditions or manifestations of a system.

The deep/causal knowledge level in MFADES contains sets of frames linked in a tree structure. These sets of frames can be considered as several modules. These modules include the following: (a) hypotheses/states frames; (b) states/states frames; (c) states/findings frames; (d) hypotheses/ hypotheses frames; (e) deep/causal rule base. In the system proposed in this paper, this architecture is defined in a frame structure and a deep/causal rule base has been defined. More details about these modules are given below.

### Hypotheses/States Frames

Certain hypotheses are defined as a presence of a particular state. In these situations, hypotheses frames in the deep/causal level are connected to states.

**States/States Frames.** This module is employed to represent cause-effect relationships between states. Two states are connected through an action. This situation is represented in the states/states frames.

**States/Findings Frames.** The states/findings frames represent the fact that a finding is an observable external manifestation of a given state.

**Hypotheses/Hypotheses Frames.** These frames represent the links between heuristic hypotheses and deep/causal hypotheses in the diagnostic system.

**Deep/Causal Rule Base.** The deep/causal rule base contains control knowledge about the structure of the frames described above and their interconnections.

### MFADES Heuristic Knowledge Architecture

Presently, MFADES diagnoses faulty behaviour of Universal Multi-Mod insertion machines by employing a heuristic knowledge module. The following sections describe the theory employed to develop MFADES architecture and an example of the diagnostics of a failure. The mathematical foundations described in the following sections have been implemented in the CHECK architecture developed by

Torasso and Console (1989).

### Mathematical Foundations of the Approximate Reasoning Schema

As mentioned above, the mathematical concepts employed to develop MFADES are discussed in Torasso and Console (1989) in Chapter 6. A brief discussion of the main concepts employed in the heuristic module of MFADES is given in this sections. For more detailed information, the reader is referred to [Lesmo et al. 1984].

The manner in which observed evidence values and relevance measures are combined depends on the logical connective employed in the complex condition. Torasso and Console [1989] report the following equations for the connective function for the OR and AND logical connectives.

$$\begin{aligned} f_{AND} &= (e, 0) = 1 \\ f_{AND} &= (e, 1) = e \\ f_{AND} &= (0, m) = 1 - m \\ f_{AND} &= (1, m) = 1 \\ f_{AND} &= (e, m) = [m * e] + [1 - m] \end{aligned} \quad (1)$$

$$\begin{aligned} f_{OR} &= (e, 0) = 0 \\ f_{OR} &= (e, 1) = e \\ f_{OR} &= (0, m) = 0 \\ f_{OR} &= (1, m) = m \\ f_{OR} &= (e, m) = m * e \end{aligned} \quad (2)$$

Equations 1 and 2 are employed to evaluate elementary conditions. However, it is necessary to obtain the revised relevance measures for complex conditions. Complex conditions occur whenever a case must evaluate more than one set of either necessary or supplementary conditions. Lesmo et al. [1984] presented the equation for the complex logical connective AND (necessary conditions) as follows:

$$e(AND(T_1, T_2, \dots, T_n)) = \alpha + \beta * (\beta - \alpha) \quad (3)$$

$$\text{where } \alpha = \prod e(T_i) \text{ and } \beta = \min e(T_i)$$

The OR (supplementary conditions) operator is given in Torasso and Console [1989] according to the De Morgan's Laws as follows:

$$e(OR(T_1, T_2, \dots, T_n)) = e(NOT(AND((NOT T_1), (NOT T_2), \dots, (NOT T_n)))) \quad (4)$$

where the semantic of the NOT operator is given in possibility theory [Zadeh, 1978] as:

$$e(NOT T) = 1 - e(T) \quad (5)$$

To obtain the combined evidence values about necessary and supplementary conditions, it is necessary to employ an equation that "favors" the operand for necessary conditions over the operand for supplementary conditions. Torasso and Console [1989] give the following equation:

$$e_1 + u e_2 = e_1 + ((1 - e_1) * e_1 * e_2) \quad (6)$$

where  $e_1$  is the evidence degree coming from the necessary conditions and  $e_2$  is the evidence degree coming from the supplementary conditions.

### An Example

As mentioned in Section 1.0, the first phase of the project consisted in developing the basic architecture of the diagnostic system and in developing part of the heuristic knowledge level. The domain knowledge employed in this project is related to failures of the Universal Multi-Mod insertion machine. The Multi-Mod machine is capable of inserting electronic components with two rows of pins in a printed circuit board (PCB) such as: integrated circuits, resistor networks, proms, sockets, etc. The Multi-Mod machine is a common machine within the corporation and suffers from a significant number of breakdowns, with few "experts" at hand capable of diagnosing and solving the problems in a speedy manner.

The failures of a Multi-Mod insertion machine can be classified the following way: (a) misinsertion; (b) transport; (c) cut and clinch; (d) program execution. Misinsertion failures occur when components are not properly inserted in the PCB. Transport failures refer to a malfunction in the machine transport mechanism. These failures can start at the magazines where the electronic components are stored, ending at the insertion head jaws. Cut and clinch failures occur when there is a malfunction in the machine and the electronic component is inserted in the PCB but is not properly secured. Program execution refers to failures in the machine program to insert components in the board.

For the purposes of this paper, a limited number of the faults of the Multi-Mod insertion machine is given. The set given corresponds to misinsertions. Misinsertion failures can be classified as follows: air blast; component alignment; body stop pin; locator pin; head stops; head rotation; lead length; head height; insertion pressure; lead sensing; anvil. These types of misinsertion failures can be further classified as shown in Figure 2. The framework shown in Figure 2 will be employed to explain the structure developed for the heuristic knowledge level.

The architecture of the heuristic level was described in Section 3. An example employing the failure modes of Figure 2 will be given in the remainder of this section. The example shows the heuristic structure for the "AIR.BLAST" failure.

A tree structure similar to the one presented in Figure 2, contains the activation rules for the heuristic diagnostic hypotheses. The activation rule for the hypothesis "AIR.BLAST" is given as follows:

#### Activation Rule AIR.BLAST

(IF (Component stability is unstable)  
THEN (consider hypothesis AIR.BLAST))

relevance: 1.0

An activation rule has an associated *relevance measure*

value. With each atomic condition in a complex condition, a relevance measure is associated within the interval [0,1]. A value close to 1 indicates that the condition has great relevance while a value close to 0 indicates that the condition has a lower relevance [Torasso and Console, 1989].

The hypothesis' frame contains information relevant to necessary and supplementary conditions, as well as specializations of the hypothesis. The hypothesis' frame for AIR.BLAST is shown below:

#### AIR.BLAST Frame

##### NECESSARY CONDITIONS

COMPONENT.STABILITY:

[UNSTABLE, 0.8 ]

[LIGHTLY.UNSTABLE, 0.7]

relevance: 1.0

##### SUPPLEMENTARY CONDITIONS

AIR.NOZZLE

[MISALIGNED, 0.7]

relevance: 0.7

SPECIALIZATIONS:

[AIR.BLAST.HORIZONTAL,

AIR.BLAST.VERTICAL,

AIR.BLAST.PRESSURE ]

An attribute in the necessary and supplementary conditions has a pair associated as shown above: [value  $V_{ij}$ , evidence value  $e_{ij}$ ]. The interpretation is that the fact that an attribute  $A_i$  takes the value  $V_{ij}$  is compatible with the hypothesis  $H$  with evidence value  $e_{ij}$  [Torasso and Console, 1989]. Notice that each attribute in the above schema has associated a relevance measure. Moreover, notice that necessary conditions are ANDed while supplementary conditions are Ored.

Following the diagram shown in Figure 2, the AIR.BLAST hypothesis has associated three specialization hypotheses: AIR.BLAST.HORIZONTAL, AIR.BLAST.VERTICAL, and AIR.BLAST.PRESSURE. The associated heuristic frames with these hypotheses are shown below. Notice that these frames are leaf nodes and contain a CORRECTIVE.ACTION slot and a DIAGNOSIS slot. These slots provide information to the operator about the causes of faults and possible corrective actions to take.

#### AIR.BLAST.HORIZONTAL Frame

##### NECESSARY CONDITIONS

COMPONENT.STATUS:

[VIBRATING, 0.9]

[LIGHTLY.VIBRATING, 0.7]

relevance: 1.0

SPECIALIZATIONS:

[ ]

DIAGNOSIS:

[AIR.NOZZLE.TOO.FAR.LEFT.OR.RIGHT]

CORRECTIVE ACTION:

[ADJUST.AIR.NOZZLE.LEFT.OR.RIGHT]

#### **AIR.BLAST.VERTICAL Frame**

##### NECESSARY CONDITIONS

COMPONENT.STATUS:

[TILTED.FORWARD, 0.99]

[LIGHTLY.TILTED FORWARD 0.7]

relevance: 1.0

SPECIALIZATIONS:

[ ]

DIAGNOSIS:

[AIR.NOZZLE.TOO.HIGH]

CORRECTIVE ACTION:

[ADJUST.AIR.NOZZLE.DOWN]

#### **AIR.BLAST.PRESSURE Frame**

##### NECESSARY CONDITIONS

COMPONENT.STATUS:

[COMPONENT.NOT.FLAT.I.N.PUSHER.SRE ADER, 0.9]

relevance: 1.0

SPECIALIZATIONS:

[ ]

DIAGNOSIS:

[AIR.BLAST.PRESSURE.TOO.LOW]

CORRECTIVE ACTION:

[INCREASE.AIRBLAST.PRESSURE]

The system evaluates a diagnostic hypothesis by employing the equations of section 4.1. To evaluate a frame, the analyst will provide information to the system about the conditions of the Multi-Mod machine. For the purpose of this example, consider that the analyst provides the system with the following information:

COMPONENT.STABILITY: *UNSTABLE*

AIR.NOZZLE: *MISALIGNED*

COMPONENT.STATUS: *TILTED.FORWARD*

THRESHOLD: 0.80

The first step is to assess if the hypothesis "AIR.BLAST" can be instantiated. This is done by evaluating the activation rules tree. Analyzing the activation rule for the AIR.BLAST hypothesis, the value of component stability is *UNSTABLE*, and therefore the hypothesis is considered for further analysis. The next step is to evaluate the NECESSARY and SUPPLEMENTARY conditions. Necessary conditions are ANDed and Equation 1 is employed for values of  $e = 0.8$  and  $m = 1.0$ . The corresponding value of  $f_{AND} = 0.8$ . Similarly, the SUPPLEMENTARY conditions are ORed and Equation 2 is employed with the values of  $e = 0.7$  and  $m = 0.7$ . The value for  $f_{OR} = 0.49$ . To obtain the combined evidence value for NECESSARY and SUPPLEMENTARY conditions Equation 6 is employed, and the final value for  $e_1 + e_2 = 0.8784$ . After this evaluation has been performed, the frame AIR.BLAST is selected for further analysis based on the fact that the corrected evidence value is greater than the given threshold (0.80). Therefore, the specialization frames to AIR.BLAST are considered as possible diagnostic hypotheses. The only frame that matches the data provided by the analyst is the AIR.BLAST.VERTICAL frame and its


corrected evidence value can be computed with Equation 1 as  $f_{AND} = 0.99$ .

At this stage, the system considers AIR.BLAST.VERTICAL as a plausible diagnostic hypothesis, and identifies a problem with the air nozzle to be too high, and provides advice for corrective action as adjusting the air nozzle down.

#### **Conclusion**

Presently, the MFADES system diagnoses failures of manufacturing equipment taking only into consideration the heuristic level knowledge explained in Section 3.1. In the next phase of the project, the deep/causal level explained in Section 3.2, will be implemented. The system will generate a possible set of diagnostic hypotheses at the heuristic level and will confirm which hypotheses are plausible at the deep/causal level. The new system will analyze all the possible connections that a hypothesis generated at the heuristic level will have at the deep/causal level. Moreover, the preventive maintenance module will be developed to generate a set of maintenance schedules for a piece of equipment.

#### **References**

- Lesmo, L., Saitta, L. and Torasso, P. [1984]. An Approximate Reasoning Framework for Expert Systems. *Proceedings of the 6th International Congress on Cybernetics and Systems*, Paris, pp:169-174.
- Torasso, P. and Console, L. (1989). *Diagnostic Problem Solving: Combining Heuristic, Approximate and Causal Reasoning*. Van Nostrand Reinhold, New York.
- Zadeh, L. A. (1978). Fuzzy Sets as a Basis for a Theory of Possibility. *Fuzzy Sets and Systems*. vol. 1, pp. 3-28. 

*Aldo Dagnino currently holds the positions of Advisory Engineer and Project Management Coordinator, at the Engineering and Manufacturing Productivity Center (EMPC) of the Transmission Technology Institute (TTI) of Asea Brown Boveri (ABB), in Raleigh, North Carolina.*

*Dr. Dagnino's role is to assist in the enhancement of the productivity of ABB plants in the Power Transmission and Distribution Business Segment, in the Americas. His mission is to introduce World Class Manufacturing approaches in ABB plants. Dr. Dagnino has extensive experience in the development of knowledge-based and object-oriented software systems to improve industrial operations.*

*Brian Schack is a Research Officer in the Advanced Computing Department at the Alberta Research Council. He has a M.Sc. in Computer Science from the University of Calgary (1988) dealing with reinforcement learning in neural nets. Currently, he is working on a project to schedule grain trucks for the Alberta Wheat Pool. He has been involved in projects ranging from automatic map interpretation to predicting hockey playoff scores to pipeline scheduling.*

# M3i Systems Inc.

*Daniel Puche*

## Introduction

Systèmes M3i inc. de Longueuil, Québec, est un chef de file mondial dans la production de systèmes logiciels intégrés pour des opérations de distribution et pour la gestion de la répartition. Nos logiciels uniques, notre expertise d'intégration et nos services professionnels améliorent l'efficacité des opérations de nos clients partout au monde. L'acronyme M3i est dérivé de l'expression anglaise: "Management through Instant Interactive Information" (gestion par information interactive instantanée), qui décrit très bien l'objet principal de notre compagnie. Nous nous spécialisons dans les systèmes d'appui pour centre de commande. Nous fournissons des solutions opérationnelles innovatrices pour les entreprises de service public d'électricité, de gaz et d'eau, pour les organismes répondant aux urgences et pour les compagnies de transport et de télécommunication.

## Introduction

M3i Systems Inc. of Longueuil, Quebec, is a world leader in producing integrated software systems for distribution operations and dispatch management. Our unique software, integration expertise, and professional services are improving operational efficiency for clients around the globe.

The acronym, M3i, stands for "Management through Instant Interactive Information," which aptly describes the focus of our business. We specialize in command center support systems, providing innovative operational solutions for electric, gas, and water utilities, for emergency response organizations, and for transportation and telecommunications industries.

## 1. Trans-Porteur Project Summary

Trans-Porteur is a research and development project for the development of interactive real-time decision-making technologies applied to the transportation sector. The research is conducted at three levels: specific particular applications within the transportation sector, real-time decision-making and generic business systems development with an object-oriented approach (corporate objects).

The Trans-Porteur project, presented to several provincial and federal funding programs, covers all the activities concerning research and development for a total of 68 million dollars spread out over 48 months. The project is presented by the lead firm for the project M3i Systems Inc., and by its partners ROADsoft Solutions Inc., APG Inc., KEOPS Informatique Inc., and Canadian Marconi.

## 2. History

The Trans-Porteur project originated from the lead partner's need to develop new technologies to satisfy the

evolutionary demands of the transportation industry in order to remain a key solutions provider. To support the necessary level of growth and development within the transportation industry, three key technologies were identified: the object-oriented approach, real-time decision-making, and intelligent control centers.

Business pressures on the transportation industry come from many sources. First, the market economy has shifted; markets are now world-wide and competition is presently between nations. Also, new products come onto the market much more rapidly and have a much shorter life span. As for manufacturers, mass production techniques which we saw in the past are now disappearing. Enterprises must now meet the conditions of personalized "made-to-measure" demand of the client. In other words, companies must do more with less. These productivity pressures increasingly push enterprises to meet these demands by relying on different and complex information technologies. The systems required must be flexible, evolutionary, and able to integrate with new technologies such as client-server architecture, object-oriented structure and virtual servers and networks.

To meet the demands of the twenty-first century, the transportation industry must have instant access to information. Consideration must be given to the complex and various sources of information, the communication infrastructures, and intelligent control systems which must be developed in order to provide carriers with the solutions necessary to respond to their client's demands. These demands of the transportation industry reflect also those of the project partners to maintain and improve their competitive positions in the market place. They will need the necessary tools for real-time decisions, secure communication links, and user-friendly evolutionary systems; in other words "made-to-measure" solutions. The Trans-Porteur project addresses the solutions to these various business problems.

## 3. The Project

The Trans-Porteur project aims to develop interactive real-time decision-making for the transportation sector. Due to the generic nature of the technologies developed in this project, these solutions could also be applied to transportation of dangerous substances, world-wide transportation, urban network management, and public transportation as well as trucking. Because the project is structured with three levels of research (Trans-LINK, d-SiD, and ISCOM), the partners will also use these technologies to serve their respective markets.

Trans-LINK (Transportation, Logistic, Integration, Navigation, and Knowledge base system) represents the highest level and addresses solutions tailored to the

transportation industry. Three principal lines of development have been identified: acquisition of information, processing of information from mobile units, and the intelligent control centre. In order to achieve error-free results, the project partners will apply their understanding of logistics control, on-board technologies, and secure communication links in their research.

The next level—d-SiD (dialogue—Status, Impact, Decision), offers a more generic tool as a development base for Trans-LINK applications. As previously mentioned, the transportation industry requires real-time direct decision-making; d-SiD responds to that need. The real-time decision-making process includes the accumulation of data and the real-time analysis of data from multiple sources to identify tendencies (impact analysis). The processed result (decision) is an optimal solution taking into consideration all data present at the time of analysis. Through available technologies such as intelligent agents, d-SiD can implement automated decision-making in areas such as administration and communications. In summary, the results will enable decision-making and not just decision support.


The main functionality of the d-SiD decision loops is to organise the decision-making process while feeding databases which will provide support for software components of the Trans-Link environment. The advantage of developing the d-SiD component is to simplify the decision-making processes by centralising the data analysis tasks, by facilitating the propagation and sharing of pertinent information within the corporate networks, and by handling the decision-making processes generated from the large quantities of information coming from real-time systems. This simplification is made possible by the automation of decision-making processes with intelligent agents capable of processing their own information, and at the same time able to communicate their internal state while acquiring from a network the corrective actions necessary for their operation. This operational decisions filtering process will allow human resources to concentrate on strategic planning, which may have far more significant impacts on corporations than the daily events which the d-SiD system will handle.

At the most fundamental level, ISCOM provides a development base for optimal engineering of information systems. Conceived around the innovative object-oriented approach, ISCOM utilises client-server architecture to make information accessible at the point of decision-making. By drawing on libraries of reusable objects during the conception and development of information systems, construction of reliable applications is accelerated to meet rapidly changing needs. Furthermore, through the development of intelligent agents, ISCOM will have the capability of bridging the gap between different corporate databases. This link will allow the inclusion of legacy systems, in which the transportation industry has already invested large amounts of money.

To understand the Trans-Porteur project in its entirety, a metaphor may be useful. It is possible to compare the project

with a truck. The technology of Trans-Porteur is like the technology used to construct a truck. Without plans or tools, assembly is impossible. ISCOM is the tool required to assemble the components (objects) and place them in an optimal manner in order to create a working system. In summary, ISCOM is like the process to assemble and maintain the truck's mechanical components. As well as mechanical components, a working truck requires elements of control—for example, electronic boxes and sensors. d-SiD technology is like the truck's electronic boxes which manage the influx of gas, air, oil, and electricity and coordinates all data to maximise the vehicle's efficiency. As for Trans-LINK, it represents the whole truck including its load. A fully loaded truck that has no driver with precise delivery instructions is useless. Like a trained and well-informed driver, Trans-LINK technology enables a transportation firm to meet its customers' expectations. In this metaphor, with all the components present, the same truck becomes an important link to the economic chain.

It is important to understand that the technologies developed in this project will have utilisations well beyond the transportation sector. APG, for example, foresees the application of ISCOM technologies within the banking sector while KEOPS will apply its d-SiD decision-making technology in control rooms of industrial enterprises. M3i will implement decision-making systems within electric services. ROADsoft Solutions and Canadian Marconi will be able to apply their diverse technologies within the trucking industry not to mention that of public transportation and road network management. The Trans-Porteur project's results have vast possibilities in a variety of commercial areas.

Transportation represents an important link in the overall economy of Quebec and Canada. The technologies from Trans-Porteur will enable carriers to perform more effectively in their markets which is a priority to remain competitive. The project will also build expertise which will be exportable world-wide. 

---

*Daniel Puche is Executive Director of the Research & Development Department of M3i Systems Inc., located in Longueuil, Quebec.*



# Generic Integrated Intelligent System Architecture for Fault Diagnosis and Maintenance

Ming Rao and Qun Wang

## Résumé

Le diagnostic de panne basé sur la technologie de l'intelligence artificielle est un domaine important des applications de l'intelligence artificielle. Les accidents et les pannes d'équipement ont toujours pour résultat la pollution de l'environnement et une pauvre qualité de produit, et mettent en péril la sécurité de l'équipement et des ressources humaines. En fin de compte, les accidents coûtent de l'argent, et ils diminuent la productivité et les profits. Dans cet article nous présentons une architecture de système intelligent intégré générique pour la surveillance d'équipement, pour le diagnostic de panne et pour l'entretien. L'architecture comprend quatre modules individuels: la calibration des données, la surveillance des conditions, le diagnostic des pannes et l'aide à l'entretien. Elle intègre diverses techniques d'intelligence artificielle comme le raisonnement heuristique, le raisonnement basé sur des cas et l'hypermedia. On présente deux applications industrielles utilisant cette architecture: "IMSS (système d'appui intelligent à l'entretien) pour la surveillance et le dépannage des camions miniers de Syncrude" et "On-line Real-time Expert System for Incident Reporting (système expert en ligne à temps réel pour le rapport d'incidents)" au moulin de pulpe de Daishowa-Marubeni International (DMI) Peace River.

## Abstract

Fault diagnosis based on artificial intelligence technology is a significant domain of AI applications. Hazardous accidents and equipment failures always result in environmental pollution and poor product quality, and jeopardize the safety of equipment and human resources. As a result, accidents cost money, and decrease productivity and profits. In this article, a generic integrated distributed intelligent system architecture for equipment monitoring, fault diagnosis and maintenance is discussed. The architecture consists of four individual modules: data calibration, condition monitoring, fault diagnosis, and maintenance assistance. It integrates different AI techniques such as heuristic reasoning, case-based reasoning, and hypermedia. Two industrial applications using the architecture are presented: "Intelligent Maintenance Support System (IMSS) for Syncrude Mining Truck Monitoring and Troubleshooting;" and "On-line Real-time Expert System for Incident Reporting" in Daishowa-Marubeni International (DMI) Peace River Pulp Mill.

## 1. Technical Development of Diagnostic Systems

Artificial intelligence has been successfully applied to

process monitoring, troubleshooting, and equipment maintenance. So far, many fault diagnosis expert systems (Kramer, 1991), intelligent monitoring systems (Murdock and Hayes-Roth, 1991), or knowledge-based maintenance systems (Berzonsky, 1990) have been developed for chemical processes, electronic devices, and mechanical equipment. These systems have shown that they can bring significant benefits to industries.

Fault or malfunction diagnosis has historically been among the most active research and application areas of artificial intelligence. The first diagnosis expert system is the MYCIN medical diagnosis system developed by the Stanford University Heuristic Project in 1972. After MYCIN was generated, many other medical diagnostic expert systems such as Casnet, Puff, PIP, and MDX were developed. Since 1985, these exploring diagnostic expert systems in turn led to the development of diagnostic reasoning techniques in a variety of engineering disciplines. So far, the intelligent fault diagnosis technology has been developed into the relative domain-independent theories of diagnosis, including many specialized diagnosis techniques for particular domains from medical science, aerospace, electronics, manufacturing, process engineering, and computer science.

Early fault diagnosis systems with symbolic reasoning was based on Boolean logic or fault trees, which used exact symbolic computation in reliability analysis mathematical models and could not deal with uncertain knowledge and information. As a result, these systems could not effectively utilize the expert's experience to handle complex real-world problem solving. MYCIN can deal with uncertain information in rule-based representation. Early diagnostic expert systems were developed using the rule-based technique. The key issue in developing a rule-based expert system is how all the components of the diagnosing system and its behaviours can be described by heuristic rules.

Even though rules are very useful to represent heuristic knowledge, they are quite poor in describing system components and functions, which are called deep-knowledge or model-based knowledge (Tou-Ng, 1991). Another disadvantage of rule-based representation is that it is very difficult to maintain, expand, and explain knowledge bases. It could result in conflicting rules and inefficient reasoning.

Most model-based diagnosis techniques and representations were built on Reiter's theory of diagnosis (Reiter, 1987) based on first principles. Reiter defined the diagnosis problem in terms of a logical triple—system description, components, and observations. Therefore, a diagnosis is a labelling of the system components as either abnormal or normal based on the system description and

observations. System description stands for system performance and specifications, while observation represents the working environment or operating conditions. Their change could result in component failure or malfunctions. Different from rule-based systems, model-based diagnosis systems focus on description of components and system performance, and on the representation between internal relations and external environment rather than directly linking symptoms to the possible underlying causes using heuristic rules. This model-based approach has some advantages over the heuristic rule approach of conventional expert systems, such as information explanation, learning-self and knowledge maintainability. Such a structural and behavioural reasoning system is necessary for diagnosing large-scale electronic-mechanical systems (Paasch and Agogino, 1993). However, building a model-based system may cost more time and money.

To utilize both advantages of rule-based and model-based diagnostic systems, many diagnostic systems combined heuristic and model-based approaches and used both reasoning mechanisms. These systems used heuristic knowledge for common faults, and first principles (model-based approach) for novel or difficult cases. This hybrid architecture greatly improved the performance of diagnosis systems and applications. As we know, the architecture is moderately successful in domains where the rule-based approach would have been difficult or impossible (for example, diagnosing novel devices).

In the past few years, many techniques have been used in developing intelligent systems for data calibration, process monitoring, malfunction diagnosis, and equipment maintenance. As a means of pattern recognition, neural networks have attracted much attention based on their capability to learn complex and nonlinear functions, which are often used to calibrate real-time data, identify poorly-understood dynamic processes, and model complex systems (Leonard and Bramer, 1993). Rule-based expert systems can be used in solving real-world problems which depend heavily on an expert's experience. However, in such expert systems, it is difficult to maintain and expand the knowledge base (Vargas and Raj, 1993). Case-based reasoning (CBR) is a problem-solving paradigm where experience is used to guide problem-solving (Stottler, 1994). Case-based reasoning uses past problem-solving cases, including success or failure results, which directly reflect the domain expert's efforts or experience. This paradigm shows a great deal of promise for use in diagnostic systems. The challenge in introducing CBR into diagnosis is how to implement reasoning of analogy, management of previous cases, and generation of novel cases. These methods have considerable merit in the situations where there is little behavioural knowledge to support rule-based or model-based approaches.

The other trend of the diagnostic system development is the use of integrated distributed techniques. The integrated techniques consist of (i) integrating different diagnostic

techniques, for example, integrated model-based and heuristic features in real-time expert systems (Pfau-Wagenbauer and Nejd, 1993) and integration of neural networks, case-based with rule-based systems for complex system diagnosis; (ii) integration of different knowledge representation such as integrating sets, rules, and data in an object-oriented environment (Czejdo et al., 1993); Combining multiple knowledge sources (Steier, et al., 1993), and integrating multiple application systems such as condition monitoring, diagnosis, and maintenance (Urnsbach, et al., 1994).

Distributed diagnostic systems are based on the distributed artificial intelligence (DAI) technology. Most researchers are investigating how a single diagnostic agent can have intelligent behaviours based on diagnostic automation. However, such a single intelligent agent is limited in solving real world problems. This is because these problems are too complex or risky to be handled by individuals. In contrast, most real world problems such as requirement diagnosis can be dealt with by a group of people who work together. In this case, one expects that multiple intelligent agents can work in an distributed computation environment to handle more complicated problems.

In this paper, we will discuss an integrated distributed intelligent system architecture for fault diagnosis. Its advantage is to integrate data processing, condition monitoring, malfunction diagnosis, and maintenance with an integrated distributed intelligent environment. The architecture employs multiple knowledge representation and sources, integrates several inference mechanisms, and runs in a real-time distributed environment.

## 2. Defining Problems in Diagnosis Systems

Process systems are subject to equipment degradation or failure, external disturbance, operator error, and inappropriate process control settings (McDowell et al., 1991). An abnormal situation is a disturbance or series of disturbances in a process that causes plant operation to deviate from their normal operating state. The nature of an abnormal situation may be of minimal or catastrophic consequence. A disturbance may simply cause a reduction in production; in more serious cases it may endanger human life (Nimmo, 1995).

Fault diagnosis for processes and equipment is quite complex and consists of data calibration, condition monitoring, detecting faults, diagnosing malfunctions, and planning corrective actions. Because each of these diagnostic activities embraces non-numeric, dynamic, uncertain, experience-based problem-solving, processes or equipment diagnosis is a significant area for applying integrated distributed intelligent system technology and distributed artificial intelligence. Investigation indicates that diagnostic systems are encountering the following challenges:

- All these diagnostic activities organize a correlated information flow. Data preprocessing, monitoring,

diagnosis, and planning corrective actions and maintenance have to be integrated for effective diagnostic problem-solving. Currently, development and application of diagnostic systems is limited to each isolated island, that is, almost all applications focus on an approach such as model-based or neural networks. As a result, they cannot be effectively utilized for a complex system diagnosis.

- Even when processes or equipment are normal, analog or digital sensors are never perfectly consistent with diagnostic models, due to modeling and measurement errors. Any diagnostic model or rule-based system cannot directly employ the original industrial data for diagnosis. The data pre-processing is necessary if the model is to be installed on a production line.
- Any single diagnostic approach such as model-based, rule-based, or neural networks is of limited application. For example, certain components are impossible to model adequately by a model-based approach. Under this situation, heuristic rules could be effective to diagnose these components from qualitative I/O states. On the other hand, for a nonlinear and dynamic process, neural networks are more robust than rule-based systems.
- Many diagnostic systems for processes are not on-line real-time expert systems even though some real-time expert systems are reported (Ingrand et al., 1992; Pfau-Wagenbauer and Nejd, 1993; Padalkar et al., 1991) For real-time diagnostic systems, there are many crucial requirements such as reasoning speed, task priority, memory management, and so on.

### 3. Integrated Distributed Intelligent System Architecture for Diagnosis

As we know, data calibration, condition monitoring, fault diagnosis, and equipment maintenance in industrial processes are serial activities. Usually, a monitoring sensor system is installed in a piece of machinery or a production line for real-time data collection. This sensor data is transferred to computer systems such as a DCS (distributed control system). Since the data involve normal, abnormal, and noise signals, it is necessary to calibrate the data and instruments. The meaningful data and information are graphically displayed on operator consoles in a control room. The data are also stored in a historical database for production management. Under an abnormal situation, the operator has to interpret the abnormal conditions that will cause an incident, determine what kinds of action will be taken, and resume the process to normal conditions. An engineer has to find reasons for equipment malfunctions based on the abnormal conditions and schedule a maintenance plan. Then a manager arranges a plant-wide production plan depending on the information provided by operators and engineers.

To realize the diagnosis activities, an integrated knowledge-based information management system is required. We propose a generic integrated distributed

intelligent system architecture for fault diagnosis. Development of the generic integrated distributed intelligent systems is based on four requirements. The first requirement is that the system is able to monitor the data, and evaluate its significance with experts' experience. The second is to develop a troubleshooting or diagnostic system to help operators. Third, a user friendly interactive information system is needed. This would include such utilities as on-line manuals and maintenance records. Finally, there is an impetus to store the knowledge of experts to prevent the loss that occurs when they leave the employment of the company. In a nutshell, it is desired to create a system that will make the work of the employees more efficient and cost effective.

This architecture integrates data calibration, monitoring, fault diagnosis, and maintenance assistance together. The data calibration module performs data-preprocessing, data compression and data format transfer. The condition monitoring module is an on-line data driven system, being similar to a forward chaining inference engine. It acts to alert the user to any alarms and their significance. The fault diagnosis unit is an interactive module which uses a hypothesis driven backwards chaining inference engine to aid the user in analyzing the defect in the machinery or process. Once the analysis is successful, the intelligent hypermedia system provides maintenance assistance to the user in remedying the fault. The storage of knowledge is facilitated by the distributed knowledge bases which are used by individual expert systems. In this methodology, the condition monitoring function acts to reduce the search space for the fault diagnosis inference engine (Ursenbach, et

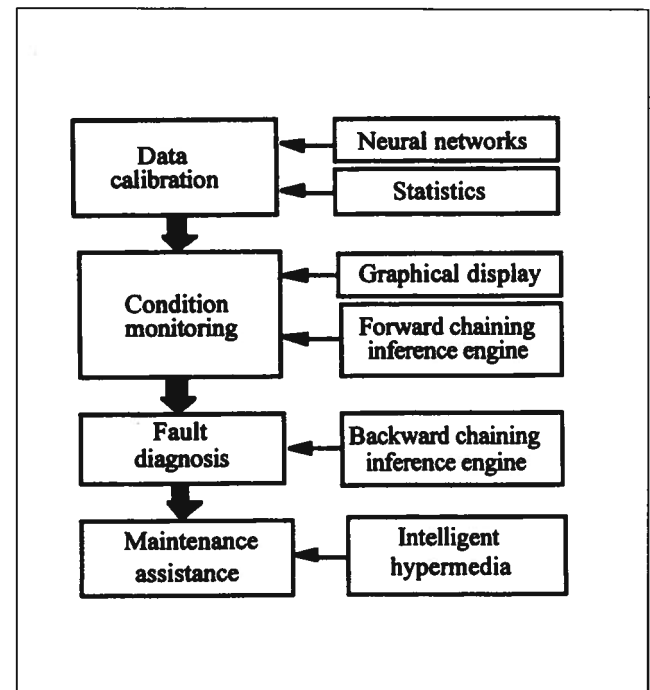


Figure 1. Integration of diagnostic activities

al., 1994). This is an effective technique, where one artificial intelligence tool is used to narrow the solution space for another (Senjen et al., 1993). In this section, we will discuss the methodology in detail; then, two industrial applications will be presented as both off-line and on-line applications.

The diagnostic modules were developed for the distribution of the following tasks: (i) data calibration, (ii) normal condition monitoring for viewing of trends, (iii) abnormal condition monitoring for intelligent interpretation of signals, (iv) component fault diagnosis for assistance in diagnosing defective components, and (v) maintenance assistance tools for providing procedural and technical information. Each module functions using information from the previous one. The practical methodology for fault diagnosis which has been developed in our technology is shown in Figure 1.

In this diagram, the primary components of the methodology are data calibration, condition monitoring, fault diagnosis, and maintenance assistance. The tools employed for the functioning of these modules are neural networks and statistics for data calibration, a graphical trend display

routine, expert systems with forward and backward chaining inference engines, and an intelligent hypermedia system. The forward chaining inference engine is used for the condition monitoring while the backward chaining inference engine is used for the fault diagnosis.

The modules work in conjunction with one another, with the results of the previous stages of operations being used in subsequent stages. The following figure details the integration of this methodology. It outlines the flow of information in the system and the maintenance process (Figure 2).

Noisy data cannot be eliminated from practical processes since there exist process disturbances and instrument calibration. Most instrumentation and computer hardware systems lack the capability to handle this issue. Soft computing based on neural network and fuzzy logic approaches could be used to solve the problem. In our integrated architecture, soft computing technology is not employed yet. Instead, we used statistical technology and practical experience to analyze original analog and digital signals. It is required to carry out data compression and data

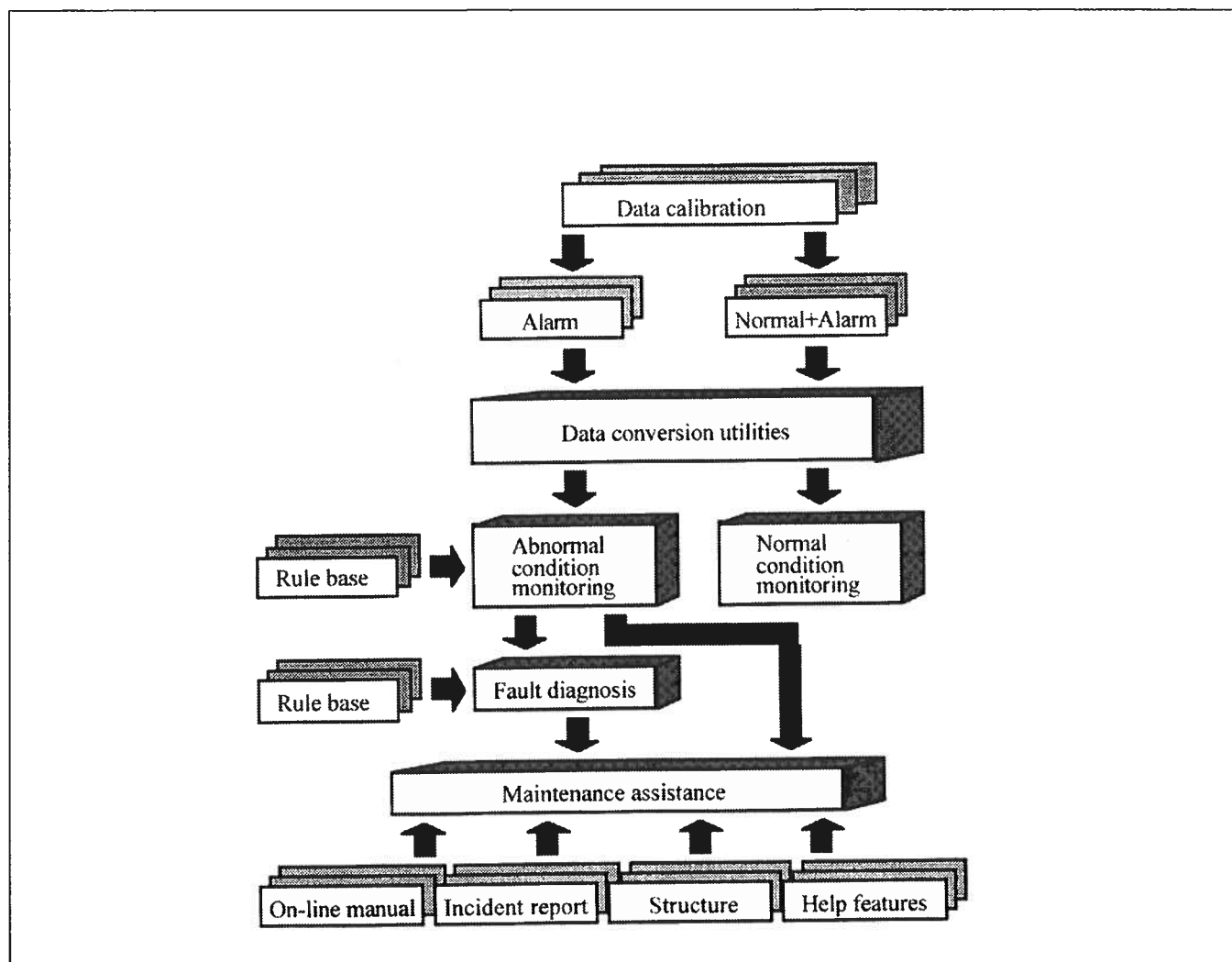


Figure 2. Overall integrated architecture and information flow.

format transfer for integration and communication among different software tools.

The monitoring activity tracks process variables intelligently and provides knowledge-based explanation of normal process behaviour, but this process is complicated by the overwhelming number of possible alarms and by the problem of alarm escalation (Dogle, et al., 1990). Fixed-threshold alarms might not be sensitive and meaningful to many process trends that can lead to the alarm state. Usually, three types of condition monitoring and alarm systems are used. The top level condition monitoring is the hardware condition monitoring. This is composed of actual sensors built into the mechanical system that monitor specific properties. The other two types of condition monitoring are software based. The normal condition monitoring is a trend display routine while the abnormal condition monitoring is an interpretive decision tool. Both use data from the hardware condition monitoring.

Under a normal situation, the normal condition monitoring module draws a curve that displays the trend of each monitored variable in the hardware condition monitoring system. When an abnormal case occurs in the equipment or process, the inference engine, using the condition monitoring rule base, provides a hypothesis of a possible problem area in which a fault has occurred or is likely to occur. Abnormal condition monitoring inference engine only deals with the data from the sensors. The abnormal condition or fault detection, which is closely related to monitoring, involves differentiating between normal and abnormal conditions. Managing this kind of problem solving requires reasoning about physical relationships in a way that explains the current process state and predicts trajectories that the process is likely to follow. Current efforts use causal models to explain alarm states and apply connection architecture to classify current process states from on-line data.

The step following the abnormal condition monitoring uses the component fault diagnostic inference engine to further analyze components in the hypothetical failure area. Malfunction diagnosis isolates and identifies process malfunctions. This can be especially difficult, given dynamic behaviours caused by control systems and by mass and energy feedback. To isolate malfunctions under dynamic conditions, we must represent and reason about feedback and effects on symptomatic information. The component fault diagnosis module uses a rule base developed specifically for failures. In providing an area of failure for the component fault diagnosis to analyze, the abnormal condition monitoring can act to reduce search space for the fault diagnosis inference engine. In practice, diagnosing the causes of a defect is a difficult undertaking. The fault diagnostics will assist in this task by asking the operators or the mechanics for more detailed information obtained by direct observation or indirect reasoning. Once all the information is gathered, the fault diagnostics then provides information concerning the state of a specific component.

When a fault has been diagnosed, the final step is to give the operator corrective actions and instructions, as well as to provide a maintenance plan to deal with the problem safely and economically. The maintenance assistance model uses the intelligent hypermedia system to implement the purpose. If a component fault has been verified, the module can automatically open the intelligent hypermedia system to display the related information and data about removal, installation, testing, and maintenance of the faulty component. It also displays information regarding the particular fault and specific procedures involved with its resolution. This is one facet of the maintenance assistance. The maintenance assistance also provides tools for browsing Mechanical system structure and process systems, case histories for troubleshooting, and on-line help functions.

Although there is a flow of information from one module to another, the distributed nature of the architecture also allows for the functioning of each component independent of the others. Any one of the modules such as normal and abnormal condition monitoring, fault diagnosis, or the hypermedia applications, may be accessed directly.

#### **4. Equipment Monitoring, Diagnosis and Maintenance for Mining Trucks**

The first application we present is a prototype intelligent maintenance support system (IMSS) for Syncrude's mining truck condition monitoring developed by the Intelligence Engineering Laboratory at the University of Alberta (Ursenbach, et al., 1994). Syncrude Canada Ltd. has the largest oil sand extraction and upgrading plant in the world. Its oil sands mine is one of the largest surface mines. At Syncrude, a truck and shovel system with 170 to 240 ton mining trucks are used in the transportation of oil sand and overburden (McKee, 1989). In 1992, the truck fleet moved approximately 636,000 tonnes of material per day. The efficiency of the truck fleet makes a notable impact on the mining productivity and the plant-wide profits (Scoble et al., 1991; McKee 1990). Mining truck breakdowns affect not only the productivity in the mining section but also production schedules of the whole plant (Donald, 1989; Malhotra and List, 1989; Jonah, 1988). As well, maintenance of the truck fleet costs several million dollars per year. Therefore, it has become necessary that the downtime of mining trucks be reduced to a minimum. Truck condition monitoring, fault diagnostics, and maintenance aids are effective tools in speeding up the maintenance process.

Due to the complexity of the large mining trucks at Syncrude, many of the trucks use a number of sensors to monitor their performance and to detect faults in various selected components. For example, on the diesel-electric 190 ton Titan T-2000™ trucks, the Vital Signs Monitor (VSM™) collects and stores 21 analog and 26 digital signals every 30 minutes, and at any time when an alarm is activated (Anon, 1992). This system of data collection generates a large number of data sets in a short period of operating time.

This data, in and of itself, is inconclusive, but is a valuable aid to operators and mechanics involved in the maintenance process.

#### 4.1 System Functions and Configuration

A summary of the functions of IMSS is listed as follows: (1) normal and abnormal condition monitoring, (2) fault diagnosis, and (3) maintenance assistance and on-line maintenance manual display. These functions are intended for use by novices in truck maintenance, truck operators, truck maintenance mechanics, and section reliability engineers at Syncrude Canada Ltd. The interface of IMSS must be easily usable by all these parties.

IMSS has therefore been developed on a Microsoft Windows™ platform. Microsoft Windows™ is rapidly becoming the defactor standard due to its ease of use and its ability to coordinate and integrate many functions. It is also the preferred platform at Syncrude which already uses a number of 486 personal computers with adequate memory. The following paragraphs describe the various functional modules accessible through the menus of IMSS.

#### 4.2 Condition Monitoring

The data from the VSM™ are handled by the normal conditioning module of IMSS. When the user selects the "Normal" option from the "Condition Monitoring" menu item in the "Troubleshooting" menu, IMSS calls the routine for plotting historical data. Figure 3 shows the screen when the trend of coolant temperature is selected.

Every time an alarm condition occurs, the VSM™ performs a data dump of all the signals from the monitored components. Each data dump may be converted into a fact base using the data conversion facilities in the IMSS. Numerical analysis packages that process the data can place their results in the fact base. The abnormal condition monitoring inference engine uses the fact base to draw conclusions regarding the condition of the machinery.

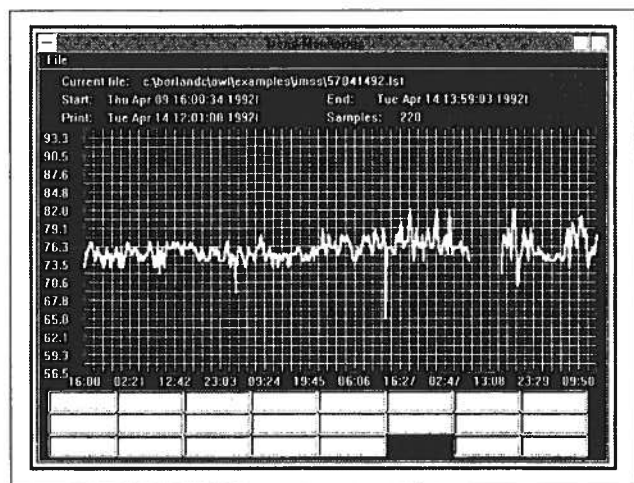


Figure 3. Coolant temperature.

A forward chaining inference engine is suitable to the task of condition monitoring as it is intended to be on-line and therefore receives information from the machine signals only (Milne et al., 1991). Requests for user input are left to the component fault diagnosis section. The VSM™ signals, stored in the fact base, are the data or facts used by the inference engine which will provide a conclusion about the state of the machinery. The VSM™ signals, however, are limited in number and generally cannot provide an exact conclusion regarding a specific component of the truck. The abnormal condition monitoring only provides a conclusion as to which general area contains the suspected fault, such as the hydraulic or mechanical systems. The component fault diagnosis further narrows the area of failure to a specific component.

At present, the abnormal condition monitoring is off-line. A session is initiated by the user from the main menu of IMSS. When the user chooses the "Abnormal" option from the sub-menu of "Condition Monitoring" in the "Troubleshooting" menu item, IMSS automatically loads the "Monitor" rule base. The forward chaining inference engine reasons through the rule base until it reaches a true condition. The abnormal condition monitoring displays the result on screen.

The menu of the "Condition Monitoring" window contains the option "Link." "Link" can call the component fault diagnosis for further analysis of the fault or it can give the user the opportunity to review a history of previous cases for that failure area. This linking facility is a part of the integration methodology for IMSS.

With the standardization of the condition monitoring formats, Syncrude may apply IMSS for use with hardware condition monitoring systems other than the VSM™. A new hardware condition monitoring system would require a new rule base and a new data translation scheme, but the basic methodology and programming of IMSS would remain the same.

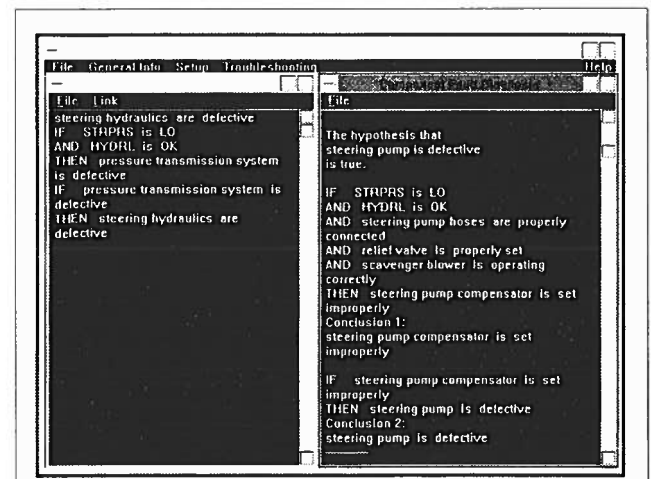


Figure 4. Display of reasoning



### 4.3 Fault Diagnosis

The component fault diagnosis module uses an expert system which reasons using VSM<sup>TM</sup> data and user input, and makes use of various hypermedia aids to provide additional information regarding the faults and user tests. The component fault diagnosis inference engine is of the backward chaining type, meaning it receives a hypothesis and verifies the truthfulness of that hypothesis against data or facts.

In the IMSS implementation, the user can put forth a hypothesis of a faulty component which the reasoning mechanism checks against the VSM<sup>TM</sup> signals and user observations. A backward chaining methodology is suitable for the diagnosis, as there are many possible facts that have to be considered in the diagnostic process, most of which are user observations and tests. It is therefore necessary to reduce the search space of the fault tree by selecting a hypothesis, which would narrow the search to the direct descendants of that point on the tree.

The component fault diagnosis module may either be called through the condition monitoring by the "Link" option, or independently from IMSS's main menu through the "Fault Diagnosis" item in the "Troubleshooting" menu. The knowledge base selected will correspond to the area of failure indicated by the abnormal condition monitoring. In the fault diagnosis process, the user enters the hypothesis of the failed component into the inference engine.

The system needs information from the user when no facts in the fact base correspond to the rule statement. If this is the case, it will bring up a dialogue box asking if the statement is true. The user may either respond with "Yes" or "No," or may request "Help". If the user presses the button for help, IMSS transfers the case to the hypermedia system, which further explains the question.

If the user answers all questions in the affirmative, and the information from the VSM<sup>TM</sup> corresponds to the conditions in the rule base, IMSS states that the hypothesis is true. It

can then display the reasoning path behind the conclusion, which is a restatement of the rules (Figure 4). The results of the test draw two conclusions. Conclusion 2 indicates which component is defective, and Conclusion 1 indicates where is the fault in that component.

From the verification of the failed component by the fault diagnosis process, IMSS has two possible routes for assistance that the user may select from the fault checking menu. They may either check the fault record, or they can use an on-line manual for information regarding the failed component in Conclusion 2. If the option for fault record is chosen, IMSS passes the fault information to the hypermedia system which will display a report on that fault (Figure 5).

The report presents a description of the normal state of the component as well as information regarding resolution for that specific fault. Alternatively, the user may call the on-line manual. When calling the on-line manual from this point in the IMSS process, the first page shown is the page describing the defective component.

There is a significant difference between these two forms of information presentation. The fault record provides information prepared specifically for IMSS and the fault diagnosis. The on-line manual simply gives general information regarding the component as described in the maintenance manuals.

### 4.4 Case-Based Reasoning

When an engineer comes across an error in a system, often the first thing he will do is to find whether this situation had previously occurred before and how it was remedied. The purpose of the incident report is to record occurrences of a fault to make it available for future review. If the incident reports are stored using the intelligent hypermedia system, they can be accessed easily by IMSS. According to the architecture of IMSS, when the abnormal condition monitoring determines a fault area, the "Link" menu gives the option to view the incident reports. If this option is chosen, IMSS brings up the incident reports for that fault area. Just as there is a fault diagnostic knowledge base for the fault area, there is a group of incident reports as well.

### 4.5 Maintenance Assistance

In an integrated system, it is useful and important to have on-line advice components to supplement the monitoring and diagnostic components of IMSS. Hypermedia is an excellent tool for this, allowing the user to browse text and graphics in any fashion or allowing the information to be accessed as necessary by the software. Having the manuals on-line allows the user to have direct and instantaneous access to the necessary page of information. IMSS selects this information intelligently from the results of the reasoning process. There are non-linear links within the hypermedia which allow the user to jump between topics instantaneously backwards and forwards. If the volume of the text is kept reasonable, down to at most one or two pages per topic, the

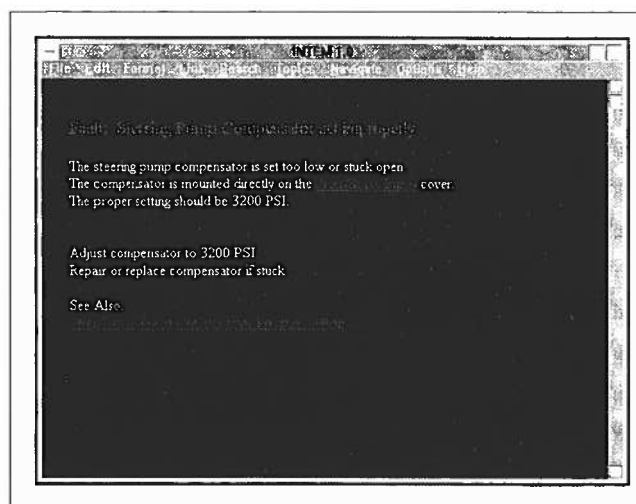


Figure 5. Fault record.

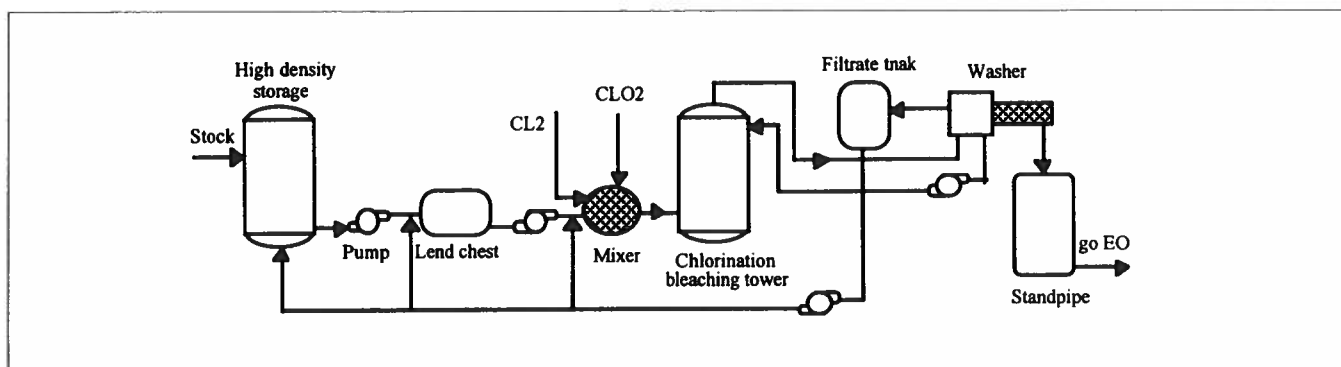


Figure 6. The CD stage in the DMI bleaching process.

hypermedia tool will be efficient time-wise and a benefit to the user (Duncan, 1993). The hypermedia system also allows for updates, corrections, and user heuristics to be placed in the manual simply by modification of the hypermedia files on disk.

Most hypermedia systems are useful in developing an information management system. They can provide information and help in using various media such as text, graphics, sound, animation, and video. To deal with real-time industrial processes and on-line information management, however, a hypermedia system must have additional capabilities. It must be able to apply formulae to real-time information, and reason from the results using human knowledge and expertise. To satisfy this need, an intelligent multimedia system for on-line real-time application (INTEMOR) was developed based on hypermedia, knowledge-based systems, and object-oriented programming in the Intelligence Engineering Laboratory at the University of Alberta (Shu, et al., 1995). Used in IMSS, INTEMOR can provide the opportunity for interactive form generation, procedural checklists, and intelligent updating of manuals and records, and allow flexibility for other needs that may arise in the future.

The manuals for the truck are being put on line. It has already been seen that the component fault diagnosis module can access the manuals, calling a specific topic from the manual related to the defective component. The user may also directly access the manual using the main menu of IMSS through the "Open Manual" item in the "General Info" menu. The manual appears in the hypermedia system open at the first page. From there, the user may browse through all available topics in the manual. The user may view the whole incident report file from the first page by selecting the "Incident Report" option from the "Troubleshooting" item in the main menu.

The user may graphically browse the structure of the truck to gain an understanding of how the various components in the system are interconnected. The purpose of this function is mainly for information and reference. Moving the mouse cursor over any of the named components and clicking on the name takes the user to a textual description of the component along with expanded graphical views. Using

this method the user will be able to navigate the various structures such as mechanical, electrical, and hydraulic structures of the mining truck.

## 5. On-line Real-time Intelligent System for Incident Prevention

Finally, we will discuss another application, an on-line real-time intelligent system for incident prevention at the DMI Peace River bleaching process. The system allows operators to respond to early warning alarms quickly and efficiently. The system not only informs operators of an impending problem with a process and the root-cause, but also instructs them on what to do before the problem becomes irreversible and causes a production upset. Before an incident occurs, the incident prevention system should compare process variables under normal and abnormal conditions, predict what will happen in the near future, provide various measures for avoiding a potential incident, and help the operator to handle the incident alarm. It must determine what the status of the potential accident is, such as occurred, immediately occurring, and warning. According to the information, the operator can choose different operation procedures.

In Daishowa-Marubeni International Ltd, for example, a fault may occur in a high inlet box pressure of one of the four C.B. bleach washers. Stated by Mr. D. Browes, an experienced operator at DMI, "if the operator does not react quickly or follow the instructions on how to remedy the incident, a production interrupt (accident) will occur. If the pulp price is about US\$750/ton (now, at US\$900/ton), and the mill produces one ton of pulp per minute, down time or production interruption 30-45 minutes will result in US\$22,500 to US\$33,750 production losses."

### 5.1 Bleaching Process Operation

The Kraft pulping is a chemical decomposition of chipped wood into fibers that can be used for making paper and paper board products. The aim of the pulping process is to separate the fibers from the lignin. For chemical pulping processes such as Kraft process, lignin is extracted by means of a chemical reaction (Reeve, 1989)

The bleaching pulp process is a very important operation to remove lignin. The brightening of pulp is done through chemical reactions, either to discolour lignin or more preferably to extract it. The common chemical bleaching processes are elemental oxygen, elemental chlorine and chlorine dioxide. For DMI Peace River Pulp Mill, its bleaching process includes chlorine/chlorine dioxide, oxygen extraction and followed by two stages of chlorine dioxide. They are called CD stage, EO stage and D1 and D2 stages. Figure 6 illustrates the chlorination bleaching stage (CD) of the DMI bleaching process. One of the greatest difficulties about operating a pulp bleaching plant is that the exact chemical composition of the pulp is unknown. Not only is it not measurable on-line by present instrumentation technology, but also it is highly variable (Erlenbach, 1995). Thus the process control is designed around affected properties that are measurable, such as brightness scale. Various on-line instruments are employed to determine measurable values for the affected properties, but by the nature of the installation it can only indicate general averaging over the pulp volume. Further, the pulp process measuring devices are extremely hard to calibrate because the principles upon which the measurement is based are typically nonlinear. A calibration is usually only accurate in a certain range of operation, with measurement degradation occurring at and past the extremes.

Another problem is that the instruments are exposed to a very rough environment. Wear and natural phenomena, such as surface coating, cause the calibrations to drift off correct settings. The control of a pulp bleaching plant thus, is plagued by the inability to measure input chemical composition, by its time-varying nature, and by the inaccurate and lacking measurements of affected properties which indicate process conditions. Precise controls are not available for a Kraft process and much of the control is left up to human know-how.

There are other difficulties inherent with the operation of a pulp bleaching plant as well. The process is very extensive, which plagues operations personnel with an over-abundance of information to deal with. A pulp bleaching process can have hundreds of process and control data values being electronically accessed. In times when operations are hectic, such as start-ups, the operations personnel cannot adequately monitor all sections of the process. During such periods, the operations personnel must focus on a few operations, leaving the other operations to do as they will. The process is realized by many physical components and devices, each having their own operating non-linearities and idiosyncrasies that need to be understood. Overall operations of the process can be confounded by abnormal events such as motors tripping off, and by operational problems such as pulp washers plugging. From the process operation personnel's point of view, there are many things that may go wrong. Corrections to a situation are not straight forward.

The process is highly interactive and what seems to be an

obvious control action often results in other unexpected problems. Operating experience over technical process knowledge is often needed to affect a reliable handling of a situation. Furthermore, when the process becomes upset, the control of the process tends to come apart and the operator is left scrambling to re-establish the process. In such duress the mind may freeze, be less effective, and be prone to making mistakes. Much of what the operator should do is overlooked in the moments of confusion. These operation difficulties are more exaggerated for new operation personnel who lack the depth in knowledge and the refinement of experience. The success of the pulp bleaching operations depends heavily on operation personnel's knowledge, ability, and the breadth of experience that they can draw on to handle the operating problems.

The monitoring and incident reporting system for the bleaching process in the DMI plant could monitor all aspects of the process, and supply well thought out situation solutions based on expertise gained through process monitoring experience. Such a system could provide twenty-four hours per day process monitoring as an operational backup aid to assist the operation personnel in noticing and handling operational situations. The potential savings in lost product, costs, and environmental issues are in the hundreds of thousands of dollars. An investment to implement a real-time intelligent system for this purpose would have the potential for very quick payback and high rate of return on the investment.

## 5.2 System Structure

The DMI mill management information system consists of DEC computers and personal computers on a computer network. The DEC computers are clustered using the network protocol DECnet™. The PCs operate under the Windows operating system and run PCSA™ for communication with DEC equipment. DEC hard disks are used for file storage.

The main computer systems and software packages consist of a distributed control system, a management information system, communication software between DEC VAX/VMS™ and PC486/MS-Windows™, as well as a real-time expert system tool and an intelligent hypermedia system, both developed by the Intelligence Engineering Laboratory at the University of Alberta. Figure 7 illustrates the hardware and software platform of our real-time expert system for incident monitoring and prevention in the mill.

The distributed control system employed by the mill is a Honeywell TDC-3000™ system, which comprises proprietary control computers (DEC), data highway, operator consoles, and so on. Real-time process data are fed into the DCS.

The management information system is called MOPS™, developed by MoDo Chemitics. MOPS collects data and information from the DCS and various mill sources, and provides the enhanced data and information to operation and management personnel. The information is presented in

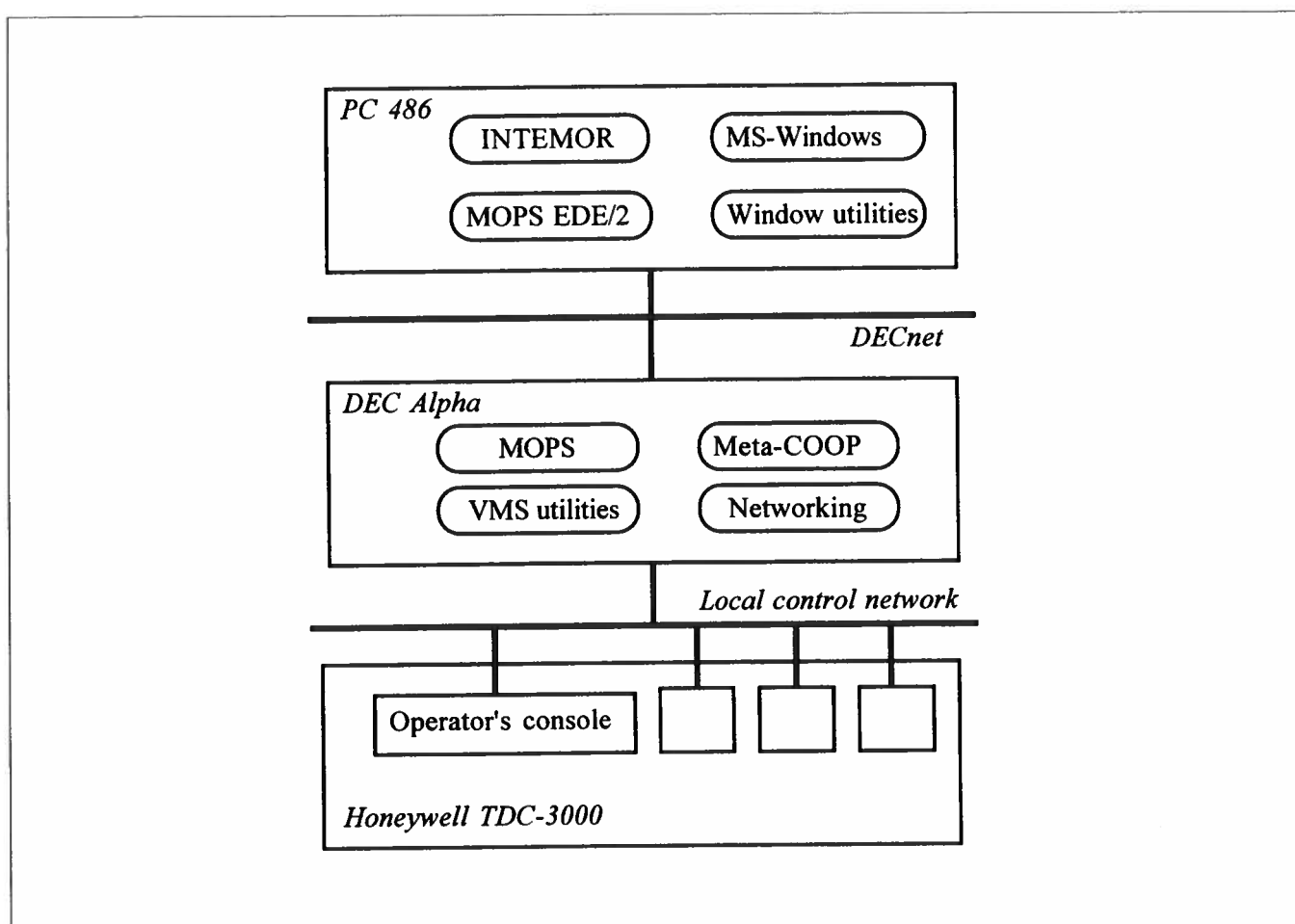


Figure 7. Software and hardware platform.

graphical forms at MOPST<sup>TM</sup> stations or as printed reports. MOPST<sup>TM</sup> compresses and stores data into a historical database (HDB) for later analysis. Current data are retained in DEC internal RAM memory in a current value database (CVD). MOPST<sup>TM</sup> offers a library of API<sup>TM</sup> routine that can be embedded in user-developed programs to access the data. MOPST<sup>TM</sup> routines execute in a DEC VAX<sup>TM</sup> cluster environment.

The communication software is MOPST<sup>TM</sup> EDE/2 that is the MOPST<sup>TM</sup> user workstation software residing in a personal computer. EDE/2<sup>TM</sup> communicates with MOPST<sup>TM</sup> via DECnet<sup>TM</sup>. EDE/2 offers graphical information such as dynamic process graphics, trends and report display. EDE/2 screens can also be configured with user input variables. Presently at the DMI mill, EDE/2 is being used for the laboratory data input. EDE/2<sup>TM</sup> workstations are remotely placed throughout the mill for process operators and management personnel.

The process data and information flow originates with the DCS acquiring field transmitter data. MOPST<sup>TM</sup> polls for the DCS data through a software communication link called Gateway<sup>TM</sup> at frequent periodic time intervals. The data is written into records in the CVD. Laboratory input and other

data are also written into the CVD. A calculation handler software process is activated periodically to perform calculations on the data. The CVD data is scanned and stored historically at intervals. For the pulp and paper industry, intervals between scans are around 30 seconds. When a user calls up a screen, a request for that screen data is sent to MOPST<sup>TM</sup>. A minimum size display information vector is sent to EDE/2 over the DECnet<sup>TM</sup>. The screen information is displayed. The display is automatically refreshed frequently while it is active.

Meta-COOP is a real-time expert system development tool (Rao, et al., 1993). Meta-COOP is coded in C, and run under the VAX/VMS<sup>TM</sup> environment. Meta-COOP provides such distinct characteristics as the integration of various knowledge representations and inference methods. The old version of Meta-COOP was designed for off-line expert system applications in a stand-alone host. To satisfy the real-time reasoning requirement in this project, we modified the system. The main modification is to enhance the reasoning speed and the memory management. In Meta-COOP, the organization structure of the knowledge-base can be divided into several components. Meta-COOP adopts the object-oriented programming technique and frame-based knowledge

representation to implement the organization, management, maintenance, and applications of knowledge. For this project, we designed several functions to carry out the communication

unique features of INTEMOR include playing multimedia information such as video, audio, and photograph files; word processing; hyper-linking and nonlinear search,

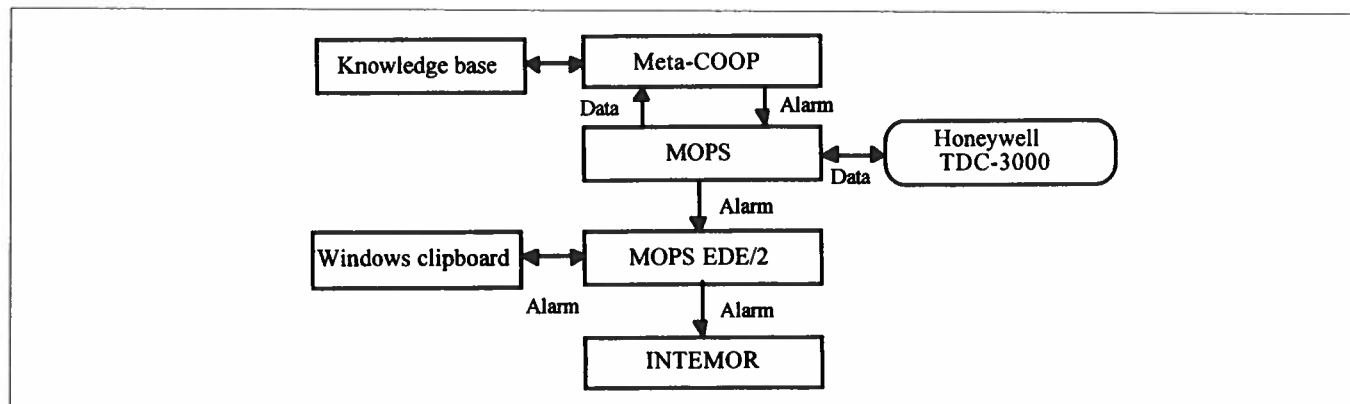


Figure 8. Components of the real-time intelligent system.

between Meta-COOP™ and the CVD (Current Value Database) of MOPS™.

The display of an incident in this project is completed on a personal computer using INTEMOR. INTEMOR runs in a PC486/66 with 8 MB RAM, and VGA with 256 colors. The

numerical computation using functions and procedures, symbolic reasoning through rule sets, and real-time data collection and manipulation.

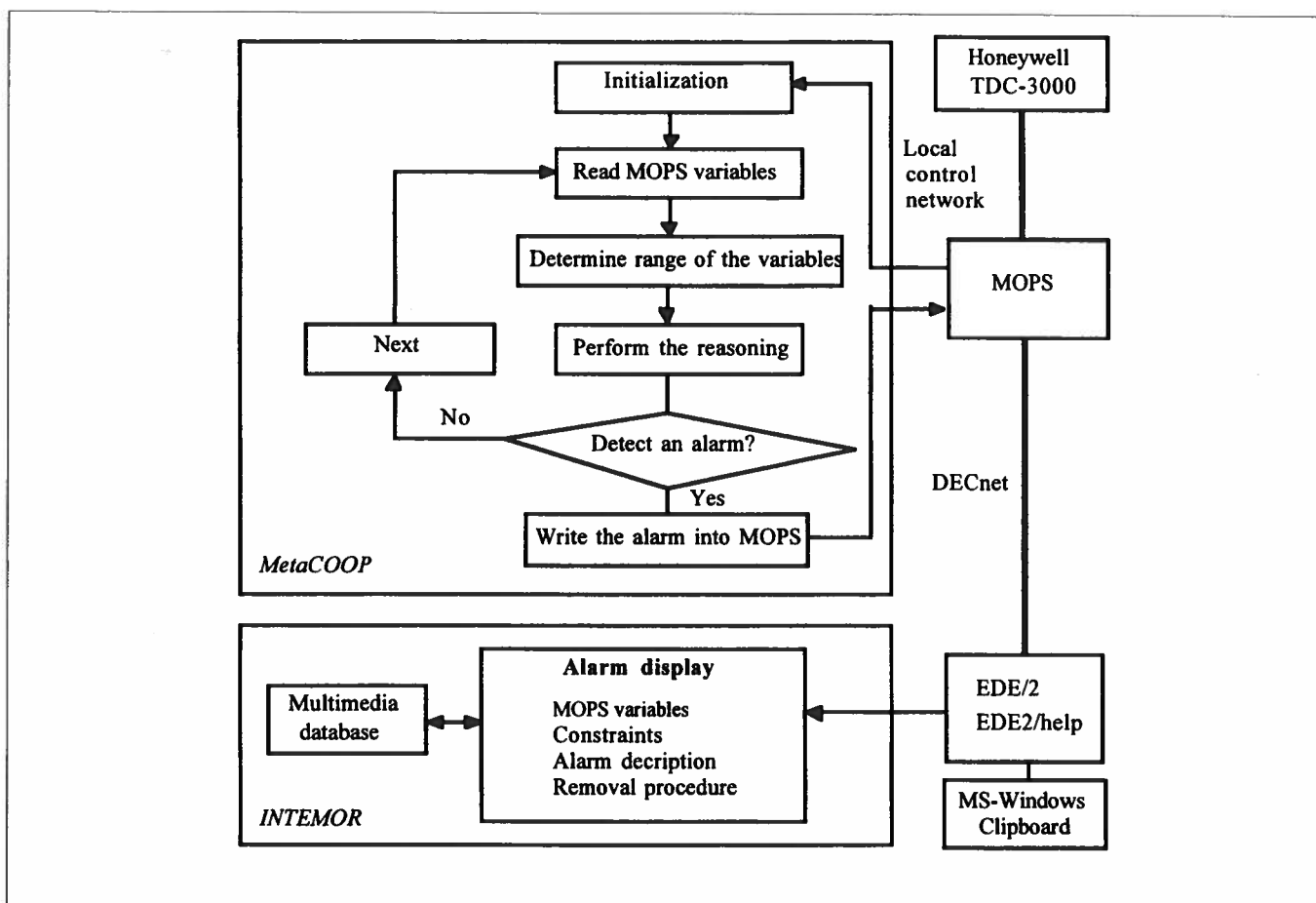


Figure 9. Information flow of the real-time intelligent system.

### 5.3 Information Flow

Figure 8 illustrates the main components of the system. MOPS™ obtains real-time process data from the Honeywell TDC-3000™ and sends control commands such as set points to the DCS (this function is now available). The real-time data will be stored in the CVD database and periodically downloaded the HDB (a history database). Meta-COOP takes these real-time data from the CVD every 30 seconds, then employs the knowledge base to deal with the data or variables. So far, the system can handle 120 process variables each time. Meta-COOP operates a very fast inference engine that only takes less than one second to scan these variables and complete a reasoning process. As mentioned above, MOPS scans and process real-time data every 30 seconds. This indicates that Meta-COOP can meet the real-time reasoning requirement and has a great potential for dealing with more process variables in a critical time period.

The results (alarms) from Meta-COOP will be put into the CVD of MOPS™. Through MOPS EDE/2™, these alarms will be written on the window clipboard. INTEMOR reads the information from the board, then retrieves a hypermedia file associated with the information. All hypermedia files

are stored in local personal computers. The process operators can navigate the selected hypermedia file to explain the alarm including causes, process components, variable values, thresholds, and procedures for removing the alarms. Here, the hypermedia files are called on-line manuals or electronic operation support manuals (Liu et al., 1995).

To explain the system in detail, Figure 9 shows the information flow of the system. TDC-3000™ is linked to MOPS™ in the DEC computer by the local control network. The expert system is also installed in the same computer. The DCL command script, STARTUP.COM, is run to start up the system. The other command script, SHUTDOWN.COM is used to stop the monitoring process.

After the STARTUP.COM is run, the main procedure of Meta-COOP will be performed. The main control diagram using Meta-COOP is shown in Figure 8.

First of all, the system needs to initialize and configure a computation environment. It also counts time, and sets up MOPS™ variables (bleach.SetMVName), process variables (bleach.SetPVName) and incident variables (bleach.SetIVName). After doing so, the program goes into a loop to monitor and report an incident. In this loop, this

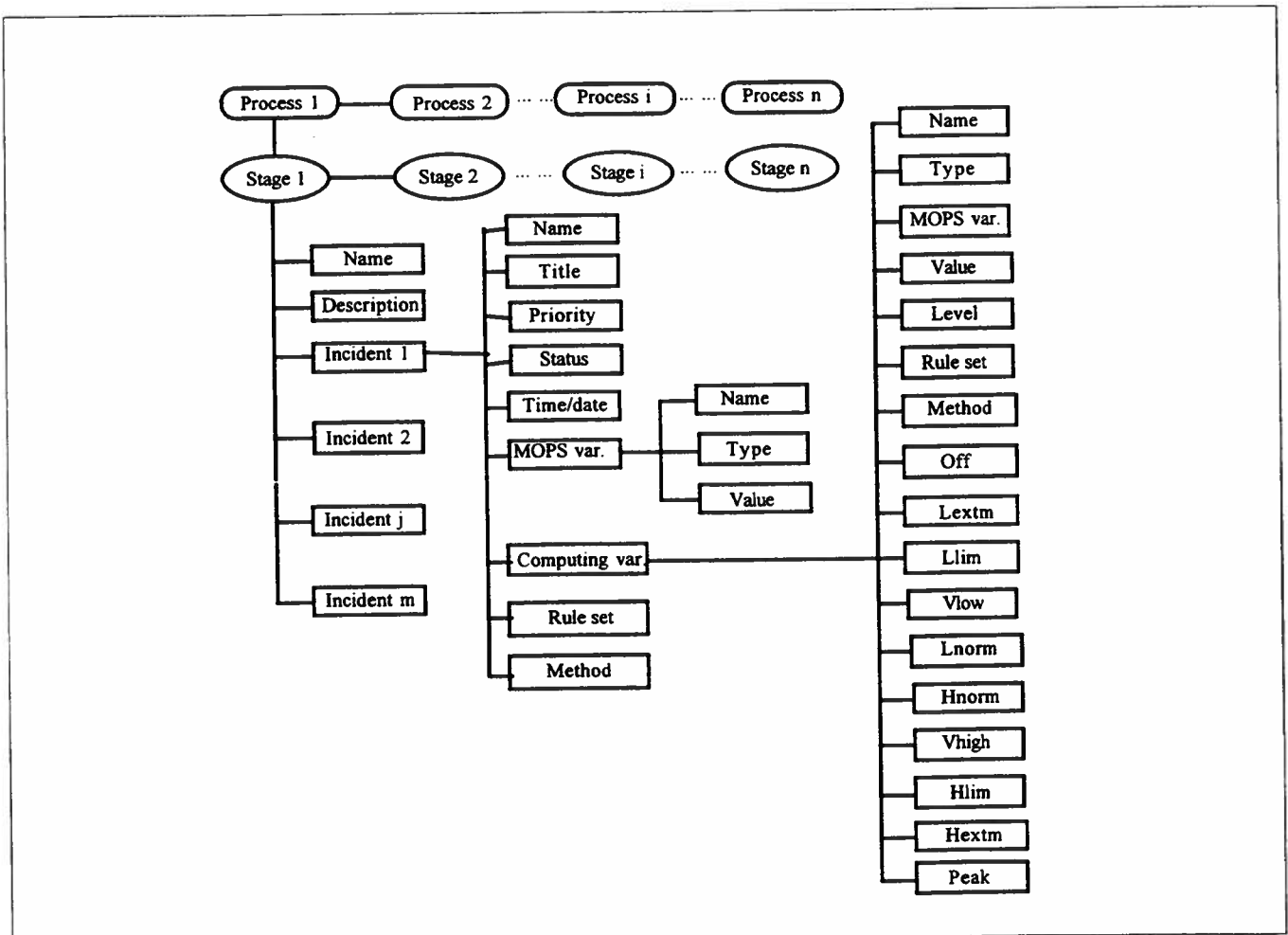


Figure 10. Knowledge representation.



system has to read the pre-defined MOPS variables using the procedure `bleach.ReadMopsData`. It should be noted that the time for scanning MOPS real-time data is every 35 seconds, and Meta-COOP has to finish data processing in this period to keep the synchrony. Currently, the system is fast enough to handle these data so that we can design a time delay function, `bleach.Time.Delay`, to keep the synchrony.

The procedures `bleach.ReadMopsData` and `bleach.ReadProcData` read the MOPS variables and set up reasoning variables. The `bleach.Set.Range` is used to set up the critical thresholds and ranges. Then the system runs the procedure `Check.Incident` to check every potential incident. Finally, the procedure `bleach.WriteToMops` writes the reasoning results and related information to the MOPS CVD™. After this loop, the system will wait for the next data processing.

Meta-COOP applies the object-oriented technology to the development of knowledge representation and construction. Usually, a knowledge base for this expert system consists of four components:

<code>bleach_v.kbs:</code>	This file defines all the global variables, variable types, classes, functions and procedures.
<code>bleach-m.kbs:</code>	All functions and procedures are declared in this file.
<code>bleach-i.kbs:</code>	All incident cases are represented in this file.
<code>bleach-r.kbs:</code>	This includes all rules used for determining the thresholds of variables.

With the OOP terminology, a superclass is a chemical process such as a digester and a bleaching process in the pulp mill. Each superclass can be divided into several stages. For example, the superclass `bleach` could include CD, EO, D1 and D2 stages. A stage could include many process variables and potential incidents. Figure 10 illustrates the knowledge base structure of the expert system. A stage is also a class that inherits the attributes and methods of its superclass.

A class often consists of several items, such as a name, description and incidents. An incident represents experience and knowledge of operation personnel, in which several components have to be organized together as follows:

Name:	an incident name
Title:	a description of the incident
Priority:	the priority of the incident
Status:	current status of the incident
Time/date:	time/date record of the incident or accident
MOPS var.:	storing the MOPS™ variables related to the incident
Computing var.:	representing the thresholds of variables
Rule set:	storing the rules that determine an incident and the thresholds
Methods:	including procedures and function

The expert system prototype in the mill contains a process, four stages and 35 incident cases. The developer interface written in Virtual Basic™ can assist the developer to set the four knowledge bases.

## 5.4 Building the Knowledge Base

As mentioned above, the knowledge base includes four components: (1) variable definition, (2) procedure and function declaration, (3) incident rules, and (4) range setting rules. The part of variable definition is described below:

After finishing the definition, the developer has to convert the quantitative expression of MOPS™ variables into the qualitative description of symbols or heuristics. This is done by three steps. First, the expert system defines the threshold values for this conversion. As shown in Figure 11, a method derives heuristic values which describe a process entities' status based on a quantified value and on an operator's expectation experience of that process entity (Sam, 1995). The expectation experience is represented as a linear range, demarcated by expectation limits (thresholds).

By comparing the quantified process values with the range limits, semantic values for several heuristic variables are determined. These values describe the process entities' status related to an operators' expectations. The specifications of the range limits are retained in a record `ProcVar`. These variables are: `off`, `lxtm`, `LLime`, `vlow`, `lnorm`, `hnorm`, `vhigh`, `hlime`, `hextm` and `peak`. `off` and `peak` demarcate the full extent of the possible values. `LLime` and `hlime` represent the maximum expected range that the process value would take. This is not an alarm limit but an expectation limit. `lnorm` and `hnorm` specify the expected range that process value would normally take. `vlow`, `vhigh`, `lxtm` and `hextm` express the extent that the process value lies outside of the normal and outside of the expected range. These limits can also be regarded as boundaries which the operator monitors for extremes in the process value which indicates drastic action is required.

Tables 1 and 2 are two examples that describe the process. Table 1 presents the CD pH process called `cdph1`. The MOPS™ variable 41-ac100241 is related to it. An incident case R930260, high pH at the CD stage, is also associated with it. All the limits have to be provided by an experienced operator. Table 2 is another example of the process of D1 tower level. MOPS™ variable 41-lc1411 is related with an incident case R930398.

In Figure 11, the expectation limits allow at least the three heuristic variables (range, level, and extent) to be concluded. The range variable describes whether the process value was found inside or out of the expected range. The level variable indicates if the process value is higher, lower, or normal as expected. The extent variable adds extra information stating whether the process value was moderate or much beyond the expected range. At least the `llime`, `hlime` and one of the `lnorm` or `hnorm` limits are required, that is, they are not null. The others could be null.

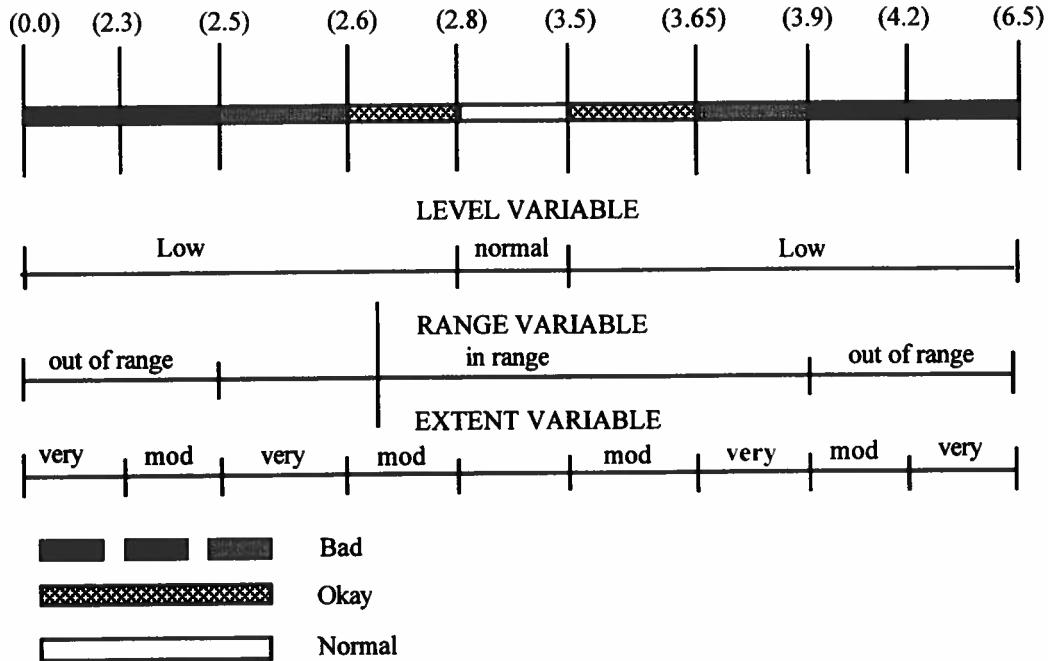


Figure 11. Expectation experience range and variable.

Conditions	OFF	LEXTM	LLIME	VLOW	LNORM	HNORM	VHIGH	HLIM	HEXTM	PEAK
xgrad=1 grade="hw"	0.0		2.0		2.2	2.8		3.8		
xgrad=1 grade="sw"	0.0		3.0		3.2	3.8		4.0		

Table 1. Limits of process (MOPST<sup>TM</sup>) variable 41-ac100241.

Conditions	OFF	LEXTM	LLIME	VLOW	LNORM	HNORM	VHIGH	HLIM	HEXTM	PEAK
xmode=1	20		73		77	87		90		100

Table 2. Limits of process (MOPST<sup>TM</sup>) variable 41-lc1411.

These limits are stored in the record ProcVar. The following two rule sets are used to convert Tables 1 and 2 into a set of rules stored in the bleach-r.bks.

After running the procedure SetRange, all the rules will be carried out by the inference engine. Then the next function will be performed, and the quantitative expression of a process variable will be converted into the qualitative description with the strings of range, level, extent. These attributes in the record ProcVar are also filled.

Let us take an example to explain the procedure SetProcVar. If the MOPST<sup>TM</sup> variable 41\_ac1002 = 4.0, that is ac1002\_41 is equal to 4.0 in the expert system, the MOPST<sup>TM</sup> variable 41\_ac1002 will be renamed in ac1002\_41. In this

case cdp11.value = ac1002\_41.value (RULE 01). Rule 01 will set up the limits if xgrade is equal to 1 and grade is HW (hard wood). These limits will be written in the instance cdp11.

```
cdp11.off:=0;
cdp11.llime:=2.0;
cdp11.lnorm:=2.2;
cdp11.hnorm:=2.8;
cdp11.hlim:=3.8;
```

After performing the procedure SetProcRange, the quantitative representation of 41\_ac1002 (41\_ac1002 = 4.0)

is converted into the description of qualitative symbols as below:

```
cdph1.range := R_OUT
cdph1.level := R_HIGH
cdph1.extent := R_VERY
```

Then, the following rule set of R930260\_R will be carried out. In this case, RULE 22 will be applicable. The status of incident R930260 is active and its condition occurs. The rule set R930398\_R will be used to determine the incident R930398 (D1 tower level is low).

Finally, the end results from the incident rule sets will be written at the variable incident[1-10] in the MOPS CVD™. For example:

INCIDENT[x]-TTL-ICD stands for the title of incident x.  
 INCIDENT[x]-TIM-ICD expresses the time/date of incident x.  
 INCIDENT[x]-STT-ICD represents the status of incident x.  
 INCIDENT[x]-HLP\_ICD stores the message of incident x that will send to INTEMOR.

## 5.5 Illustration

In this section, we will discuss the integration of Meta-COOP with the millwide system and the interface for the operators, and illustrate the end result of the previous discussion.

It is required to integrate Meta-COOP into the Daishowa plant computer system and data flow environment for developing a real time process monitoring expert system. In this case, Meta-COOP has to access data in the MOPS CVD™. The procedure ReadMopsData is used to perform the task. The MOPS function getcvd is embedded in the procedure.

The user interface transfers results from the expert system to the operator. The user interface is closely tied in with the MOPS system. When a process monitoring result has occurred, the expert system writes an alarm signal into the MOPS, which will be transferred to the DCS. An audible alarm is sent out and an alarm message appears on the console, which indicates that there are new expert system results on the MOPS™ workstation. A dedicated key is configured for the MOPS™ workstations that brings up the incident summary screen (screen showing expert system results).

This EDE/2 screen lists recent incidents and summary information. This information includes the title of the incident, time of occurrence, status of the incident and whether the incident is active or not. The active or inactive state of the incident is depicted by a box beside the title. The box is displayed in red if the incident is active, or green if inactive. The status specifies whether the incident is seen as

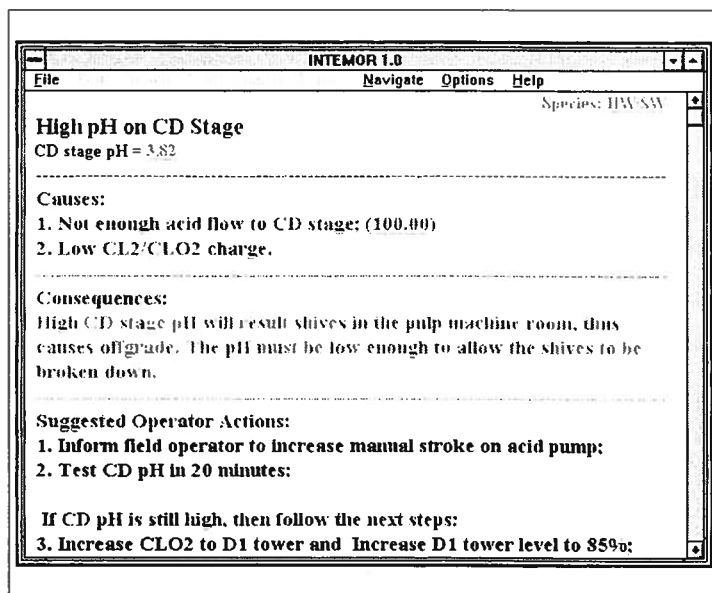


Figure 12. INTEMOR hypermedia information screen.

a warning of possible occurrence, a warning of imminent occurrence, or an alarm for the incident. For this system implementation, the ten most recent incidents are shown.

All the screen information comes from the MOPS CVD™ where the expert system writes the results. The operator can select an incident title on the screen, and then, call up the associated INTEMOR screen by clicking a menu box. The intelligent hypermedia system is automatically initiated if it is not presently running. The operator can see the detailed textual information about the cause, possible consequences, and preferred actions that should be taken (a hypermedia incident information screen can be seen in Figure 12). Also on the screen is a snap shot of selected relevant process data from the time the incident was reasoned to be active. This data provides the operator with a means to verify the expert system results. The INTEMOR display can have hyperlinks to other topics (screen) that supply further information. The screen may display equipment operating procedures, or mill equipment tagout locations. The displayed incident knowledge is edited separately from the Meta-COOP knowledge base configuration.

INTEMOR was integrated with MOPS EDE/2™ to provide a user-friendly interface. Most of the integration development work was performed by MoDo Chemetics; however, routines were written by our laboratory to fetch data for INTEMOR. When more information about an incident is needed at the EDE/2 display, a request for the CVD data for that incident is sent off to the VAX™. The VAX resident MOPS™ sends back the information in the form of text record. The record includes the name of the INTEMOR executable program, the INTEMOR document file name to be opened, the incident name, and data in the form of the variable name followed by the value (see Figure 13). The topic name for an INTEMOR screen is given the same name as the experience object (incident) in the expert system.

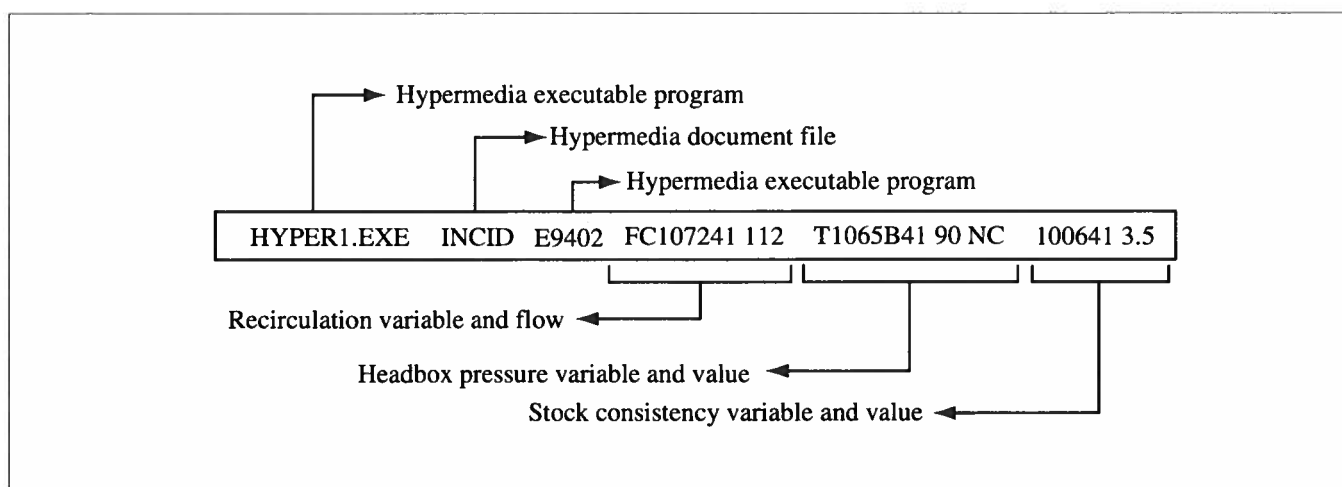


Figure 13. INTEMOR communication record.

EDE/2 captures this returned record and writes it into the Windows clipboard™. A terminate-and-stay-resident program is initiated, which parses the program and document file name, and activates the program in the Windows foreground. INTEMOR, when called forth, executes a routine that reads the clipboard and parses the rest of the message. The appropriate INTEMOR display (topic) is summoned and data is written to the matching variables. The user now is viewing the hypermedia screen.

## 6. Conclusions

It is very important and useful to implement integration of fault diagnosis with data calibration, condition monitoring, troubleshooting and maintenance. The proposed generic architecture for this purpose has been used in two industrial applications.


## 7. Acknowledgments

The author would like to acknowledge the following organizations: Syncrude Canada Ltd., Daishowa-Marubeni Inc., MoDo Chemetics, Ranger Slave Lake Pulp, Perde Enterprise and Canada-Alberta Partnership in Forestry Program for their continuous financial support and technical aid to the project. The authors would like to thank NSERC (Natural Sciences and Engineering Research Council of Canada) for important support through its research grants to the first author. Thanks are also due to our laboratory graduate students and research staff (A. Ursenbach, S. Erlenbach, H. Yang, Y. Shu, Y. Ying and others) who participated in the projects and made important contributions.

## References

- Anon (1992) Titan T-2240 truck-maintenance manuals, *Marathon LeTourneau Company*.
- Berzonsky, B.E. (1990) A knowledge-based Electrical Diagnostic System for Mining Machine Maintenance, *IEEE*

- Transaction on Industry Applications*, 26(2), 342-346.
- Czejdo, B., C.F. Eick and M. Taylor (1993) Integrating Sets, rules, and data in an object-oriented environment, *IEEE Expert*, February, 59-66.
- Dogle, R.J., S.M. Sellers and D.J. Atkinson (1990) A Focused, Context-Sensitive Approach to Monitoring, *Proc. 11th Int. Joint Conf. Artificial Intelligence*, Morgan Kaufmann, Calif. 1231-1237.
- Donald, D. A. (1989) Long term mine planning at Syncrude, *International Journal of Surface Mining*, 3, 27-34.
- Duncan, R. (1993) Building the basic structure of a Windows Help file", *PC Magazine*, May, 349-354.
- Erlenbach, S.W. (1995) *A Knowledge Representation Architecture for Expert Process Monitoring Systems: Applied to a Pulp Bleaching Process*, M.S. Thesis, University of Alberta.
- Jonah, K. (1988). Syncrude's mine production planning, *Mine Planning and Equipment Selection*, 443-455.
- Ingrand, F.F., M.P. Georgeff and A.S. Rao (1992) An Architecture for Real-Time Reasoning and System Control, *IEEE Expert*, December, 34-44.
- Kramer, M.A. (1991) First International Workshop on Principles of Diagnosis, *IEEE Expert*, June, 86-88.
- Leonard, J.A. and M.A. Kramer (1993) Diagnosing Dynamic Faults Using Modular Neural Nets, *IEEE Expert*, April, 44-53.
- Liu W. and Q. Xia (1995) *Intelligent Multimedia On-line Manual Development at the Ranger Slave Lake Pulp Corporation*, Proceedings of the Second Intelligence Engineering Laboratory Research Review Conference, 235-244.
- Malhotra, Y. P., and List, S. J. (1989) Short term mine planning at Syncrude, *International Journal of Surface Mining and Reclamation*, 3, 11-20.
- McDowell J.K., M.A. Kramer and J.F. Davis (1991) Knowledge-Based Diagnosis in Process Engineering, *IEEE Expert*, June, 65-66.

- McKee, W. N. (1989) Overview of the Syncrude mining operation, *International Journal of Surface Mining and Reclamation*, 3, 1-9.
- McKee, W. N. (1990) Technical support to the Syncrude oilsands mine, *International Journal of Surface Mining and Reclamation*, 4, 83-7.
- Milne, R., Bain, E., and Drummond, M. (1991) Predicting faults with real-time diagnosis, *Proceedings of the IEEE 30th Conference on Decision and Control*, Brighton, England, 2598-2603.
- Murdock, J.L. and B. Hayes-Roth (1991) Intelligent Monitoring And Control of semiconductor Manufacturing Equipment, *IEEE Expert*, December, 19-31.
- Nimmo, J. (1995) Abnormal Situation Management: Giving Your Control System Ability to 'Cope'", *Honeywell Journal for Industrial Automation and Control*, 2(2), 5-6.
- Paasch, R.K. and A.M. Agogino (1993) A Structural and Behavioral Reasoning System for Diagnosing Large-Scale Systems, *IEEE Expert*, August, 31-36.
- Padalkar, S., G. Karsai, C. Biegl, J. Sztipanovits, K. Okuda and N. Miyasaka (1991) Real-Time Fault Diagnostics, *IEEE Expert*, June, 75-85.
- Pfau-Wagenbauer, M and W. Negdl (1993) Integrating Model-based and Heuristic Features in a Real-time Expert System, *IEEE Expert*, August, 12-18.
- Rao, M., Q. Wang and J. Cha (1993) *Integrated Distributed Intelligent Systems for Manufacturing*, Chapman & Hall, London.
- Reiter, R. (1987) A theory of diagnosing from first principles, *Artificial Intelligence*, 32(1), 97-130.
- Reeve, D.W. (1989) *Pulp and Paper Manufacture: Volume 5: Alkaline Pulping* (T.M. Grace & E.W. Malcolm eds), CPPA, Montreal, Canada.
- Senjen, R., de Beler, M., Leckie, C., and Rowles, C. (1993) Hybrid Expert Systems for Monitoring and Fault Diagnosis, *Proceedings of the IEEE 9th Conference on Artificial Intelligence for Applications*, Orlando, FL, USA, 235-241.
- Scoble, M. J., Peck, J., and Hendricks, C. (1991) A study of surface mine equipment monitoring, *International Journal of Surface Mining and Reclamation*, 5, 111-116.
- Shu Y., H. Yang and H. Qiu (1995) *Intelligent Multimedia System for On-line Real-time Application*", Proceedings of the Second Intelligence Engineering Laboratory Research Review Conference, 76-94.
- Steier, D.M., R.L. Lewis, J. Fain and A.L. Zacherl (1993) Combining Multiple Knowledge Sources in an Integrated Intelligent System, *IEEE Expert*, June, 35-44.
- Stottler, R.H. (1994) CBR for cost and sales prediction, *AI Expert*, August, 25-33.
- Tou-Ng, H. (1991) Model-based, Multiple-Fault Diagnosis of Dynamic, Continuous Physical Devices, *IEEE Expert*, December, 38-43.
- Ursenbach, A., Q. Wang and M. Rao (1994) Intelligent Maintenance Support System for Mining Truck Condition monitoring and troubleshooting, *Intern. J. of Surface Mining, Mining, Reclamation and Environment*, 8, 73-81.
- Vargas, J.E. and S. Raj (1993) Developing maintainable expert systems using case-based reasoning, *Expert Systems*, 10(4), 219-25. 

**Dr. Ming Rao** received a B.Sc. in Chemical Engineering, an M.Sc. in Computer Science, and a PhD. in Interdisciplinary Engineering. He is a professor at the University of Alberta, and Director of the Intelligence Engineering Laboratory of the University.

**Dr. Qun Wang** received a B.Sc. in Mining Engineering, and M.Sc. and PhD. in Mechanical Engineering. He was a post doctoral fellow at Intelligence Engineering Laboratory at the University of Alberta during 1991 - 1995.

### Announcing

## The Seventh International Workshop on the Principles of Diagnosis: DX-96

Val Morin, Quebec, Canada  
October 13 - 16, 1996

Sue Abu-Hakima, Co-Chair  
E-Mail: suhayya@ai.iit.nrc.ca  
or check conferences under [http://ai.iit.nrc.ca/SPIN\\_public](http://ai.iit.nrc.ca/SPIN_public)



### Sixth annual IRIS/PRECARN conference

*Harry Rogers*

#### **Sixth Annual IRIS PRECARN Conference a Success**

"The best yet" seemed to be the general thrust of comments about this year's sixth annual IRIS PRECARN Conference, held from June 4-6 in Montreal, Quebec. A little fun was thrown in with the very serious business of presenting the results of our research in intelligent systems technologies.

This year's theme, Formula 1: Technology + Timing + Team (chosen because the Montreal Grand Prix was being held the week after our event), added some colour and creativity to our sessions and social events. At the opening reception, delegates not only had the opportunity to view over 30 demonstrations and some 90 student poster panels, they also enjoyed a casual "race-track" ambiance.

Presentations at both the opening Plenary (by Dr. Max Cynader) and Closing Banquet (by Major General (ret.) Lewis MacKenzie) emphasized the importance of the three "T"s by citing real-life experiences, from the challenges of launching a new company to keeping a team together under extremely difficult circumstances. The audience was entertained by very dynamic presentations, while being provided some valuable "lessons learned" from the experiences of our two keynote speakers.

The opening Plenary ended on a high note, with the announcement of five new PRECARN projects (see page x), announced by the Chair of the Board of Directors, Mr. Bernie MacIsaac.

The theme continued to play an important role throughout the Conference with panel sessions on specific projects (technology), on commercialization (timing), and on working in collaboration with other partners (team). Delegates noted an increased emphasis on issues related to commercialization, such as IP protection, generating business cases and finding sources of financial support. These panel sessions highlighted key issues faced by university and industry researchers, and were presented in an effort to assist those researchers and their graduate students to find success in their efforts.

As in past years, the Conference included a mix of panel sessions, mini workshops, and research paper presentations. Over 85 people had the opportunity to present the latest developments within their research projects.

One of the highlights of this annual event is the awarding of prizes. This year, in addition to the student poster prizes (see page x), the demo prize (see page x) and the Scholarship winner (see page x), we were very thankful to have attracted the generous support of several corporate sponsors. Door prizes, coffee breaks, and after-dinner chocolates at the Closing Banquet were all possible because of the generous contributions of companies. It was great fun to see the winners of the door prizes stand up and scream "IRIS" or

"PRECARN" in order to claim their prizes.

The success of such an event is always largely dependent on the many volunteers who help the staff put it together. A special thanks to all of the presenters, the session chairs, and to the students from McGill University who donated their time and talents to contribute to the success of the event.

#### **Planning for the Future**

For the past few months, the PRECARN staff and Board of Directors have gone through an extensive exercise of planning for the future. As a result, we now have comprehensive Strategic, Commercial and Operational Plans to help guide our activities for 1996 and beyond.

In developing the Plans, a few key areas of activity were identified for this year. In 1996, PRECARN will devote more time to technology management and marketing, from identifying and attracting new private sector investors to carrying-out an assessment of the commercial potential of our IRIS research projects.

IRIS has recently gone through a second-year review process. As a result of that exercise, the Network was restructured. Related to the second-year review, IRIS published a 1995 Annual Report. In that report, we highlighted some of the Network's success stories. For a copy of the IRIS report, please contact Lise McCourt at the address noted below.

#### **Feasibility studies for long-term projects**

Of PRECARN's 15 new Phase 2 projects announced last June, five fell into the long-term (up to four years) category. The feasibility studies for these projects began soon after the announcement and will wrap up in June. Here's a look at the work that is under way.

##### **Smart sensors**

An innovative research project is focusing on the use of intelligent sensing systems for the oil and mining industries. The successful application of these systems to resource industries has one of the highest potential leverages for the economy. It is estimated that improvements of only one per cent in oil extraction/processing translates into tens of millions of dollars in annual economic return.

Partners in the ISOM (Intelligent Sensing systems for the Oil and Mining industries) project are the Alberta Research Council, Syncrude Canada Ltd., MPB Technologies Inc., the University of British Columbia, and the B.C. Advanced Systems Institute. The project will break new ground in "sensor fusion" — the use of a combination of sensors along with appropriate system intelligence.



The feasibility study will establish the most efficient sensor fusion approach to four intelligent sensing systems:

- the on-line sensing of oil in aqueous slurries such as oil sands tailing streams
- an on-line system for sensing of oil contained in flows from high water cut wells
- the remote sensing of bitumen content in oil sands mining
- the remote sensing of the grading of ore in a hard rock mining operation.

"We want to build on existing sensor technology and come up with novel techniques to get the information we need," explains project leader George Sedgewick from the Alberta Research Council. "We're pushing the technology into new areas."

#### **Planning for the worst**

The SIPO (System for Intelligent Process Operation) project is building on the success of another PRECARN project—and acronym!—APACS. SIPO is a computer-based system that assists an operator controlling an industrial process plant for optimum production or during an upset or emergency. The project involves Ontario Hydro, CAE, AECL, and the University of Toronto.

"APACS is a diagnostic tool that tells an operator what's wrong, how the plant is degrading," explains project leader Jordan Chou from Ontario Hydro. "SIPO will help an operator decide what to do about it and plan a course of action to achieve the operational objectives."

Such a system is becoming increasingly necessary as industrial plants become more complex. When faced with a problem, operators must choose an appropriate course of action that best satisfies multiple conflicting goals. The goal of the feasibility study is to demonstrate the technical feasibility of the technology and the economic benefits of SIPO's planning and prediction capabilities.

"SIPO will have to evaluate alternative scenarios of future plant operation," adds Chou. "That's a very difficult task and the real technical challenge in this project."

#### **Climbing the walls**

The goal of the ACROBAT project is to make sure the project lives up to its name. ACROBAT (Autonomous Climbing Robots and Associated Technologies) is all about building a climbing robot. Its principal application: inspecting the piping in the lower feeder cabinet of a CANDU nuclear plant—an enclosed, highly radioactive space. The work builds on the technology, equipment, and expertise developed under PRECARN's ARK project, as well as that of the academic participants.

ACROBAT team members are from AECL, Inukturn Services Ltd., McGill University, Ontario Hydro Technologies, University of British Columbia, University of Toronto, and York University.

"Our first task in the feasibility study is to make sure we understand in considerable detail the geometry and physical environment in which the robot will be operating," says project leader Dr. Bruce Nickerson from Ontario Hydro Technologies. "The more we learn about the problem, the

more we'll learn about the solution.

"Secondly, we have to make a business case for ACROBAT. Right now we know that Ontario Hydro would purchase several such robots and use them very effectively. However, that's not enough. We have to design something that can not only be immediately deployed in an Ontario Hydro nuclear plant, but also has other markets in other industries."

#### **Making better decisions**

*Training machines to "think" like users* — that's the goal of the AADT project. (The acronym stands for Application of Advanced Database Technologies to Dynamic Evaluation of Situation Models).

"We're proposing to find better ways to support decision-making where there's a large amount of data that changes with time and is incomplete, inconsistent, or ambiguous," explains Phil Gray, engineering manager, Aviation, for MacDonald Dettwiler and Associates (MDA), the project leader.

"Conventional approaches force users of large databases to try to form mental models of the data. We're doing the opposite—putting what normally goes on in a decision maker's head into a machine. Our interest is in presenting the data in ways that more closely reflect the user's cognitive process and models. The result should be more effective decision-making."

MDA's partner in the AADT project is Simon Fraser University, with support from the Defence Research Establishment Valcartier and ESSA Software Ltd.

One of the goals of the feasibility study is to identify application areas. Options include military situation management, environmental management, and air traffic flow management.

#### **"Reflexe" action**

The objective of the REFLEXE (Robotics Engineering for FLEXible Environments) project is to automate more of the tasks normally performed in teleoperation—the manual control of remote robots. Most current systems lack the dexterity, responsiveness, and intelligence required to be cost-effective or even to carry out some essential tasks. This is where REFLEXE will make a difference—to develop new technologies that will lead to faster and more reliable control of teleoperated robots. The main task of the feasibility study is selecting the appropriate technologies and algorithms.

The partners in REFLEXE are Hydro Québec's research institute (IREQ), the Canadian Space Agency, McGill Centre for Intelligent Machines, the Artificial Vision Laboratory of l'Université Laval, and two Montréal-based technology companies—TECOM Systems and Machina Sapiens. The team is actively seeking another industrial partner.

Technologies emerging from REFLEXE will be applied in hostile environments that jeopardize the health of human

*continued, page 59*



## BOOK REVIEWS

**A symbolic and connectionist approach to legal information retrieval** Daniel E. Rose (Apple Computer Inc) Hillsdale, NJ: Lawrence Erlbaum Associates, 1994, xvi+314 pp; hardbound, ISBN 0-8058-1388-8, US\$59.95

Reviewed by  
Judith Dick

*A symbolic and connectionist approach to legal information retrieval* (SCALIR), was written as a doctoral dissertation at the University of California at San Diego. In this revised-for-publication edition, the story has been fleshed out, but the dissertation format is still apparent. Rose was demonstrating his ability by constructing a hybrid AI system that deals with a real-world problem, performing legal research tasks. He intended to improve upon the WESTLAW and LEXIS legal information retrieval systems. SCALIR is a hybrid system that combines connectionist processing in a self-organizing network with marker passing, and a symbolic representation that is like a semantic net.

SCALIR's combination of connectionist and IR (information retrieval) techniques is also interesting, and it is an idea whose time has come. Both techniques are probabilistic, both are concerned with efficient and speedy computing, and both have the potential to deal with large volumes of information. Where IR is a vintage technique, connectionism is a newer development whose promise has yet to be fully realized. Moreover, with the continuing interest in circumventing natural language analysis by use of statistical methods, it is surprising that we have not seen more work of this type.

The database consists of the U.S. copyright statute and cases on copyright law published by West to the end of 1988. West Digest topics are included as are Shepard's Citations. Both publishers contributed directly to the project.

SCALIR is composed of fully recurrent nets processing vector values and having multiple link types. It is unlayered. Rose's doctoral advisor was Rick Belew, who wrote AIR (Adaptive Information Retrieval). SCALIR is a direct descendant of AIR, and includes a similar learning algorithm, which employs simple reinforcement, making small changes to the net by using presynaptic activation and postsynaptic feedback. The changes adapt the representation over time to optimize retrieval by constructing paths through the database and additional links. Finding documents that an individual will judge relevant to his or her stated query is a different sort of challenge for a neural net. Relevance is not like convergence to a globally minimal error.

There are many conceptual threads in the discussion of SCALIR. It is a pleasure to read about interdisciplinary research. The project and the book appear to be work-intensive. However, narrower and deeper investigation of some of the ideas Rose touches upon might have produced a more rewarding result.

Dr. Rose is clearly a man of the marketplace. His interest is in what will work now and he is presently employed by Apple Computers. He has addressed the needs of legal information users as he perceived them to be in legal practice. He is interested in volume use, and wants his system to answer the commonly asked questions, and to optimize retrieval by using learning.

Rose was impressed by the realization that documents — law cases and statutes — are the real stuff of the law, its artifacts, not just reports, as are the documents of some other disciplines. He regards full text as the real knowledge representation. However, SCALIR's representation combines terms, subject headings, and citations, the usual stuff of manually analyzed text.

The major problem in legal research, as he perceives it, is the location of relevant precedents. The search for precedents is but one aspect of legal research. In so narrowing his focus, he has limited the scope of his vision. It suits his purpose to attempt document retrieval as opposed to the retrieval of information, that is, answers to questions.

For his subject analysis, Rose depends on the West indexing structure. The West index is a part of the symbolic representation. He is an avowed admirer of the keynumber system. The occasional anomalies, idiosyncrasies and ambiguities one finds in using West's classification scheme go without comment. He accepts it as a tried-and-true analysis of law in the area, rather than as a place to start researching a problem, as an experienced searcher might have done. There is an occasional indication that Rose has been influenced by the Critical Legal Theory people. The representation is coarse-grained, which although useful, has its limitations.

The background chapters contain an extensive survey of the literature, particularly the IR literature, demonstrating a comprehensive knowledge of the issues in theory and system evaluation. Some of Rose's summary accounts are accurate and perspicuous; others are almost true. Occasionally, he slips off the edge of an idea. All summaries are assiduously referenced.

Rose is properly appreciative of relevance feedback and vector searching, and appropriately skeptical of relevance-based valuation. He prefers noise to precision as a selection

device, as it is more sensitive. Being based on size, it can be adjusted at will.

Occasionally, there is a spicy, thought-provoking generalization, and surprisingly, these are accompanied neither by references nor by justifying arguments. These may be post-thesis ideas, as yet not fully mature. It is too bad. There is a summary discussion of semantic nets, marker passing, and knowledge representation, as well as of legal literature and research and later of some connectionist issues.

Following the background chapters is a description of SCALIR and its workings. The copyright cases, a database provided by West Publishing Company, were indexed statistically in a reasonable manner yielding an average of ten terms per document. Both documents and terms then became nodes with bidirectional links.

There is a brief discussion of the statute analysis, which deals with the logic of section drafting. He discusses the if-then structure apparent in many statute sections as making them "roughly equivalent" to symbolic rules. Although Layman Allen's work is mentioned, it appears that the logical analysis of individual statute sections was limited, as his main concern is the construction of the statute nodes. He does appear to have used statute structure and section citations as aids in indexing as has been done successfully in the past. SCALIR's nodes, about 500 in number, contain documents both statutory and case, and terms.

The term nodes are related to the document nodes by 'C-links' (connectionist links). In this way, the documents, are directly retrievable. The C-link has an adjustable weight, which can be altered as experience is accumulated. Note that one may retrieve an entire case to browse, and the text of the documents is not manipulated in any way.

In order to improve upon accessibility, the West Digest topical headings, are incorporated as S-links (structure links). The links are labelled to enable selection in searching. Rose views the structure as an approximation to a semantic net. He speculates that it is possible to perform inference in it. The weights of the S-links do not change as the structure of the subject analysis, the hierarchical index trees, remain static.

Shepard's citations, showing which cases cited which precedents and statute sections and in what manner, are the basis of H-links (hybrid links). Like the S-links, they are labelled for selective searching. The information represented by the H-links is compressed into two dimensions, one for factual information, the other for conceptual legal information. Rose says that it is possible to perform inference using the labelled H-links, but does not demonstrate that use. He wants to limit propagation to transitivity. It is a bit difficult, without some examples, to see how these H-links are actually working since the citators comments are terse and not uniformly applied. Shepard's is somewhat more critically reviewed than was West, although Rose does not address the usual points of concern. The H-links, like the C-links, have adaptable weights which may be changed over

time with use.

Rose argues that keywords and phrases are powerful expressions of meaning because they are names, tags, or labels. He relies on a functional interpretation of Wittgenstein to justify this argument as has traditionally been done. He does not deal with the larger issues of expressing the meaning of text, other than to say that at the present time AI and natural-language representations are impractical. That statement, of course, also expresses a common sentiment. Like many before him, he makes no attempt to make use of any of the analytic techniques which might be helpful when applied separately. He states his goal is to make an "end run" around the natural-language problem.

The symbolic representation of the S- and H- links is intended to enhance the meaningfulness of the representation. The subject hierarchy (S-links) aids in distinguishing senses and clarifying ambiguities. The H-links specify connections among interrelated cases, largely on the basis of the decisions, and indicate as well some cases that are distinguished. West digests and Shepard's citators provide access to some of the paper trails to precedents Rose is anxious for SCALIR to follow.

It is hoped that even though the query representation is limited to unrelated uniterms, processing connections in the combined representation will result in effective case retrieval. It is an attempt to isolate relevant documents by means of a calculus of the contexts in which the documents are classed, rather than trying to cope with the analysis of text for meaning.

The calculus of contexts is dependent on sub-symbolic processing. SCALIR's representation is localist and heterogeneous. It is a self-organizing system of the type that works well in noisy complex systems. It is unsupervised, as the IR problem is not so simple that it can be characterized by a sample of queries.

Relevance is not a binary decision, Rose postulates, it is directly proportionate to the degree of activation. The representation is continuous-valued. All documents in the database are relevant to the query to some degree. In attempting to improve upon "brittle" systems like WESTLAW and LEXIS, instead of the binary decision in a character-by-character match where a single deviation can result in a negative judgement, SCALIR's matching process is continuous.

As a result of this arrangement, the representation gradually decays. Rose considers "graceful decay" a distinct advantage. All documents in the database are relevant, it's just a matter of degree, and there is greater consideration of 'almost' matches.

The user activates the network by listing uniterms to describe his or her query. In the test examples, they were entered using dummy nodes in the interface. It was not clear whether or not the order of entry is controlled or has an effect. SCALIR automatically adjusts for plurals but makes no other morphological changes—no stemming, elliptical or wildcard devices. It provides access to ideas by means of lexemes only. To achieve the initial optimization, queries

were iterated five times, with increasingly accurate results.

The input nodes are mapped to output nodes (those with the highest activation ultimately). Matching occurs along the vectors. The process works by reorganizing the net according to features chosen by the input data. The functions are dependent on data modulation, affected by the nature of the input, and selection is performed on labelled links. It is the classificatory power of the labelled links that enables the filtering of information.

A neighbourhood is defined by the highly activated nodes. It is argued that documents in the vicinity of relevant documents are more likely to be relevant themselves than documents at a distance. Rose compares the activity to sound waves emanating from a bell — the further away from the origin of the sound, the less vibration there is. Clustering has been under investigation for some decades in automatic indexing with varying results. The SCALIR approach has a different, perhaps stronger, analytic but is still dependent on the strength of the relationship between the keyword and the central node.

Rose says that by adding the West digest terms, the database was greatly expanded. At that time, he had to investigate an earlier narrowing of the search in order to avoid too widespread activation. He developed a method which he describes as a beam of light which helped to focus the search. For the serious researcher, interested in fine tuning, the narrowing of the search early on is a matter of concern.

In SCALIR, it is possible through an accumulation of activation from variant paths to build up a high level of activation — and so to uncover, or determine relevancy in, additional nodes at some distance from the ones directly affected. Where the contexts interact, it is argued that the polysemy problem can be reduced. Presumably, he is relying on the subject headings to help disambiguate the index terms, for example.

The spread of activation is limited by restricting the amount leaving any node to a maximum of one. The S-links do not have adaptable weights and remain unaltered as they represent the subject classification structure. The subjects must maintain their relations to one another. They sustain one-third of the activation leaving a node. Individual S-links may have different weights for different nodes if there are varying numbers of outgoing S-links since the total sum of activation for the S-links leaving a node must be one-third of the activation. The C- and H-links compete for the remaining two-thirds.

Rose assumed that only about two dozen cases are useful in response to an average query. He gave users only the top-ranking documents in each search. The user always gets something back, since all the documents are relatively relevant. Obviously, the quality of system response varied from query to query.

User relevance judgements in the tests were made by individuals upon receipt of output. User feedback was solicited and entered in the form of lists of nodes to be

expanded, and nodes to be pruned.

Rose's idea is to optimize retrieval by accumulating users' experience, following AIR's development of the adaptive retrieval technique. SCALIR's representation changes in response to user feedback. The learning algorithm is the simple reinforcement calculation used by Belew, with a couple of modifications. Rose chose it because it is a heuristic that "seems to produce the desired result" (p. 213).

Nodes marked for expansion are rewarded with higher weights, and those marked for pruning suffer reductions. The rate of learning is arbitrarily set and Rose suggests a range of low values for the learning factor. The idea is that experience in using the database should have a reliable cumulative effect. If the learning factor is set too high, then both instability from occasional dramatic results, and general unreliability from small sample use may occur. The learning algorithm also constructs C-links wherever a connection is determined by other means than term-document relationship, and a C-link is absent. In so doing, it augments the ten-term-per-document indexing factor automatically. In these two ways, SCALIR adapts the representation gradually, as it learns.

Rose speculates on using SCALIR in law firms of from one to fifty partners. Within a single firm there will be specialists in different areas, users with variant levels of expertise and points of view, all of which will affect searching. The usual way of reducing the effect of such variations is through the use of large user samples. One wonders if even in a very large law firm, the population of users is large enough to normalize usage statistics. Also, it seems likely that the order of input of the adapting data would affect the status of the net, especially if the amounts were small.

Even if the adaptive changes work, is the application really appropriate? Finding the paths to the documents considered relevant to the common questions would produce easier access to the frequently used documents. Charting highways through the database may highlight the often visited nodes and obscure the relevance of documents in less frequently used nodes. It seems inappropriate to limit the ground of research to the commonly used sources, except for a preliminary survey. It is conceivable that SCALIR could have a distressingly controlling effect on research.

SCALIR is said to scale up well. Continuous matching can be projected accurately for large volumes. It would seem that adding material is relatively simple, but it does require significant statistical adjustment, moving Rose to recommend that it be done only occasionally. This is bad news, given the importance of currency in law. Also, although Rose assures us there is still plenty of room in SCALIR, it represents only a small amount of law. Representing all the law may take some doing. Recall that the addition of the West subject heading nodes caused him to develop the beam to narrow the search.

Providing a good interface was another of Rose's goals. He followed Shneiderman's recommendations for a clear

simple interface and was reasonably successful. Also, the browsing capability is good. Panning and zooming are not possible. Occasionally, testers complained about the complexity of the linkage maps, and the shading out of low activation nodes toward the edges of the screen. Probably, presentation of a good overview perspective would improve the situation.

Rose attempted three evaluative tests: quantitative, qualitative, and comparative. None of them can be said to have been well controlled or executed. He simply tried to do too much too quickly. However, attempting the tests attests to his good intention. The quantitative test, done using the traditional IR measures of precision and recall, was based on a very small sample of users. He was unfortunate in being unable to attract the number and kind of subjects he needed for any of the tests. His questionnaire research was disappointing in that users were unable to complete the forms in the time allotted.

Finally, his third test, comparing the performance of SCALIR to WESTLAW, constituted a modified success. A "stripped down" SCALIR — one in which search procedures could be readily compared with WESTLAW — was pitted against WESTLAW and held its own, with a success rate just slightly higher than that of WESTLAW. Both systems retrieved irrelevant cases. SCALIR's failures were acknowledged to involve most often its inability to accommodate string searches of any kind. The learning algorithm functioned efficiently, adapting the system to retrieve phrase subjects such as "fair use" and combinations of "data" and "base" and some synonyms. Finally, there were failures to retrieve recognized relevant documents by both systems — the most worrisome outcome given the importance of recall in legal research.

It is interesting to note how much of the users' time was spent trying to replicate features from the boolean systems. Their comments reflect their reliance on the features of LEXIS and WESTLAW, and the difficulty they had comprehending the unfamiliar connectionist model.

Rose exults over SCALIR's ability to function more flexibly than the boolean systems. However, it is clear that SCALIR's capability is limited, especially with regard to the use of language and the meaningful expression of ideas.

SCALIR appears to be a good design for a fast, efficient document retrieval system. It can make a first cut at finding information on a subject in U.S. copyright law by checking digests and citators.

Rose's claim that it scales up well bears further examination. We have no real test so far. Whether or not it contributes significantly to AI depends on one's view of what AI is. The objective of making an end run around the natural language problem has not been met.

The degree of success that Rose has achieved depends upon one's expectations. If an efficient, up-to-date document-retrieval system is what you want to sort through volumes of cases, then SCALIR may be a benchmark system. But, if what you would like to see emerging in AI research is

a conceptual system that would respond to questions with meaningful information, then SCALIR will disappoint you.

---

*Judith Dick's PhD research at the University of Toronto was in AI-based methods in retrieval of legal texts. She has many years of experience as a legal researcher.*

**Reasoning about knowledge** Ronald Fagin, Joseph Y. Halperin, Yoram Moses, and Moshe Y. Vardi (IBM Almaden Research Center, the Weizmann Institute of Science, and Rice University) Cambridge, MA: The MIT Press, 1995, xiii+477 pp; hardbound, ISBN 0-262-06162-7, \$US45.00

Reviewed by  
Peter F. Patel-Schneider  
AT&T Research

Reasoning about knowledge is concerned with answering questions like "What do I/you/we know?," "What do I/you/we *not* know?," "What do we all commonly know?," and "Do I/you/we know enough to . . ." The book *Reasoning About Knowledge* presents and analyzes a formal treatment of this reasoning about knowledge held by groups of agents. (The book uses the currently-overused term 'agents,' so I will also.) The authors are among the foremost experts in the area and are responsible for many of the recent advances in reasoning about knowledge. *Reasoning About Knowledge* is not a general-purpose book about knowledge representation and reasoning — it sticks entirely to its one portion of this broad area.

Even though the book covers only one small portion of knowledge representation and reasoning, it is nonetheless an important book. Reasoning about knowledge is an important portion of knowledge representation and reasoning, being of interest in cryptography, theory of economics, game theory, and philosophy, as well as artificial intelligence. Further, reasoning about knowledge underlies all of communication, as communication is an attempt to make someone else know (or believe) something. Communication vitally depends on what the participants know about each other's knowledge. Therefore, reasoning about knowledge is a vital skill for all intelligent agents.

The book treats reasoning about knowledge largely as a branch of formal logic, and, moreover, it emphasizes perfect reasoning, which is not appropriate in many contexts. However, an understanding of the simpler, perfect reasoning case is needed, in my opinion, as a precursor to the more complex, limited or imperfect reasoning case, and the book does devote considerable space to some formal models of the more complex case. (Whether *formal* treatments of the imperfect case are appropriate is a separate matter; even if you think such formal methods are garbage, it is generally a good idea to know something about rival camps.)

The book starts with an in-depth analysis of the formal underpinnings of reasoning about knowledge. The formal model used in the book for reasoning about knowledge is a modal logic with many modal operators, at least one per agent, given a semantics via Kripke structures. In most of the book, the possibility relations in the Kripke structures are equivalence relations and the underlying logic is standard propositional logic. This is a suitable formalism for many problems involving reasoning about knowledge, the book argues, but it has certain problems, including its tie to perfect reasoners and its limited expressive power. A more complete treatment of reasoning about knowledge in more expressive logics would be welcome, but there are probably still too many unresolved issues there for a book treatment.

The book goes on to analyze the propositional version of this model, providing a sound and complete proof theory and algorithms for reasoning about knowledge. An analysis of the complexity of various variants of reasoning about knowledge are also given. Later in the book, a temporal dimension is added to the model, and the complexity of reasoning here is also analyzed.

There is an analysis of common knowledge, i.e., knowledge that is known in common by all agents, in the book. Common knowledge is vital to agreement — if common knowledge cannot be obtained then agents cannot agree to any coordinated activity because some of them might then be missing knowledge and might not participate. Unfortunately, common knowledge is extraordinarily difficult to obtain. For example, if communication is uncertain, then common knowledge is impossible to acquire. This is the paradox of common knowledge; common knowledge is required for coordination and agreement but common knowledge is impossible to obtain in practice. The book defines some formal approximations of common knowledge that are slightly easier to obtain and are adequate for some purposes. The book, however, does not investigate the everyday way of overcoming this paradox, which is to just assume common knowledge and to be wrong sometimes.

One problem with the formalism used in most of the book is that it assumes that all reasoners are logically omniscient. This is not always realistic, as actual systems are time-limited and often have faults. The book analyzes some formal systems, variants of the above scheme, that weaken reasoning to inference in weaker, non-standard logics; that weaken reasoning by allowing impossible worlds; that weaken reasoning by restricting it to consider only atoms or terms that the reasoner is aware of; or that identify reasoning with the results of some particular algorithm.

On a more practical side, the book defines and investigates multi-agent systems. A multi-agent system is a formal system that contains several processes that each hold some local state and can access only this state. A multi-agent system also has an environment that stores the rest of the information required by the system. The agents in a multi-agent system can communicate with one another and with the environment and can perform actions that affect the

environment. Multi-agent systems can be used to effectively model knowledge bases, game trees, message-passing systems, and a host of other interesting systems. The formal mechanisms described in the book correspond to these multi-agent systems and can be used to analyze them.

The agents in a multi-agent system perform actions under control of some protocol, which is a (potentially non-deterministic) rule for selecting actions based directly on the local state of the agent. Even with this direct method for selecting actions, multi-agent systems with very simple rules can exhibit complex behavior. The book presents a generalization of multi-agent systems where the rules can refer to the knowledge of the agent and not just the explicitly-stored local state. Being able to refer to the (implicit) knowledge of the agent allows for very compact descriptions of very complex behavior, and makes it much easier to describe such complex systems. However, an implementation of such a knowledge-based system must provide an equivalent direct system and there may be many such equivalent systems, or even none, for example if the rules are self-defeating. The book analyzes conditions that guarantee the existence of a unique equivalent direct system.

One problem with agents that can use their implicit knowledge is that this implicit knowledge may be difficult to compute. The book defines multi-agent systems where the agents can act only on the knowledge they can compute, by means of some (time-limited) algorithm that is included in the agent's description. This kind of knowledge-based multi-agent system is more realistic but potentially much harder to analyze.

The book contains many interesting and useful examples, including the muddy children puzzle, the coordinated attack problem, and the stock trading paradox. These examples help to explain and clarify many of the more difficult concepts related to reasoning about knowledge. The book is targeted towards course use, but would be useful for anyone with an interest in reasoning about knowledge. Some background in formal logic is required to understand the formalisms in the book, but any serious student of artificial intelligence should have sufficient knowledge of formal logic. There are a large number of exercises suitable for classroom use or self-study in the book.

In summary, this is an extremely well-written book that covers a subject that is important to artificial intelligence. (The only really annoying point about the book is its overuse of acronyms.) I recommend it for anyone interested in problems that occur when one agent must reason about the knowledge of another, as well as anyone interested in analyzing systems consisting of multiple communicating agents.

---

*Peter F. Patel-Schneider is a Member of Technical Staff in the Artificial Intelligence Principles Research Department at AT&T Research in Murray Hill, New Jersey. His recent research has concerned description logics and description logic-based knowledge representation systems. Patel-*

Schneider's address is: AT&T Research, Room 2A-404, 600 Mountain Avenue, PO Box 636, Murray Hill, NJ 07974-0636, U.S.A.; e-mail: [pfps@research.att.com](mailto:pfps@research.att.com)

## BOOKS RECEIVED

Readers who wish to review books for *Canadian Artificial Intelligence* should write, outlining their qualifications, to the book review editor, Graeme Hirst, Department of Computer Science, University of Toronto, Toronto, Canada M5S 1A4, or send electronic mail to [gh@cs.toronto.edu](mailto:gh@cs.toronto.edu) or [gh@cs.utoronto.ca](mailto:gh@cs.utoronto.ca). Obviously, we cannot promise the availability of books in anyone's exact area of interest.

Authors and publishers who wish their books to be considered for review in *Canadian Artificial Intelligence* should send a copy to the book review editor at the address above. All books received will be listed, but not all can be reviewed.

**Meta-logics and logic programming** Krzysztof Apt and Franco Turini (editors) (University of Amsterdam and University of Pisa) Cambridge, MA: The MIT Press (Logic programming series, edited by Ehud Shapiro), 1995, xv+339~pp; hardbound, ISBN 0-262-02252-2, US\$40.00

**Computers and thought** Edward A. Feigenbaum and Julian Feldman (editors) (Stanford University and University of California, Irvine) [First published in 1963] Menlo Park, CA: AAAI Press and Cambridge, MA: The MIT Press, 1995, xiv+535~pp; paperbound, ISBN 0-262-56092-5, US\$18.00

**Android epistemology** Kenneth M. Ford, Clark Glymour, Patrick J. Hayes (editors) (University of West Florida, Carnegie Mellon University, and University of Illinois) Menlo Park, CA: AAAI Press and Cambridge, MA: The MIT Press, 1995, xvii+316~pp; hardbound, ISBN 0-262-06184-8, US\$25.00

**Language industries atlas** Paul M. Hearn and Diana F. Button (editors) (INK—Belgium) Amsterdam: IOS Press, 1994, xviii+406~pp; hardbound, ISBN 90-5199-148-7, no price listed

**Text Encoding Initiative: Background and context** Nancy Ide and Jean Vèronis (editors) (Vassar College and CNRS and Université de Provence) [Reprinted from *Computers and the Humanities*, volume 29, numbers 1, 2, and 5, 1995.] Dordrecht: Kluwer Academic Publishers, 1995, vi+242~pp; hardbound, ISBN 0-7923-3689-5, US\$99.00

**General MIDI** Stanley Jung Leib Madison, WI: A-R Editions Inc (The computer music and digital audio series, edited by John Strawn), 1995, xix+233~pp and diskette; paperbound, ISBN 0-89579-310-5, US\$27.95

**Elements of machine learning** Pat Langley (Institute for the Study of Learning and Expertise and Stanford University) San Francisco: Morgan Kaufmann Publishers, 1996, xii+419~pp; hardbound, ISBN 1-55860-301-8, US\$44.95

**Computation and intelligence: Collected readings** George F. Luger (editor) (University of New Mexico) Menlo Park, CA: AAAI Press and Cambridge, MA: The MIT Press, 1995, xii+735~pp; paperbound, ISBN 0-262-62101-0, US\$35.00

**Mind as motion: Explorations in the dynamics of cognition** Robert F. Port and Timothy van Gelder (editors) (Indiana University and Australian National University) Cambridge, MA: The MIT Press, 1995, x+590~pp; hardbound, ISBN 0-262-16150-8, US\$60.00

**The generative lexicon** James Pustejovsky (Brandeis University) Cambridge, MA: The MIT Press, 1995, xii+298~pp; hardbound, ISBN 0-262-16158-3, US\$35.00

**Goal-driven learning** Ashwin Ram and David B. Leake (editors) (Georgia Institute of Technology and Indiana University) Cambridge, MA: The MIT Press, 1995, xxii+507~pp; hardbound, ISBN 0-262-18165-7, US\$55.00

**CSCL-95: Computer support for collaborative learning** [Proceedings of The First International Conference on Computer Support for Collaborative Learning, Bloomington, Indiana, 17—20 October 1995] John L. Schnase and Edward L. Cunnius (Missouri Botanical Garden and Washington University) Distributed by Lawrence Erlbaum Associates; paperbound, ISBN 0-8058-2243-7, US\$39.95

## PRECARN UPDATE (con't. from page 53)

operators. Two applications will be worked on: the maintenance of high-voltage lines and the maintenance of structures in space.

We're being careful not to limit the outcomes to a single large product, says project leader Martin Boyer from IREQ. We want to have several smaller products that the two smaller companies and the additional industrial partner in our project could add to their product lines.

More information on the REFLEXE project can be found on the Web at <http://www.robot.ireq.ca/Projects/REFLEXE/REFLEXE.html>.

**Harry Rogers is President and CEO of PRECARD and Director of IRIS.**

For more information on any of the above feasibility studies, please contact **Lise McCourt** at (613) 727-9576 or [mccourt@precarn.ca](mailto:mccourt@precarn.ca).



## CSCSI/SCEIO Membership

I wish to join CSCSI/SCEIO and receive Canadian Artificial Intelligence

☐

• Web Access Only (\$30.00\* Cdn./yr.)

☐

• Printed Copy and Web Access (\$40.00 \*Cdn./yr.)

I am a student

☐

• Web Access Only (\$15.00\* Cdn./yr.)

☐

• Printed Copy and Web Access (\$15.00\* Cdn./yr.)

I am a member of CIPS

☐

• Web Access Only (\$25.00\* Cdn./yr.)

☐

• Printed Copy and Web Access (\$30.00\* Cdn./yr.)

Name

---

Mailing

Address

---

---

E- mail

Address

---

---

Please mail your membership to:

CIPS

430 King Street West, Suite 106

Toronto, Ontario

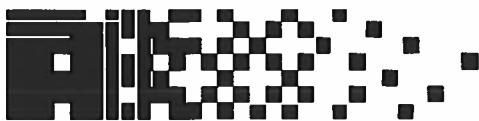
M5V 1L5

Phone: (416) 593 - 4040

Fax: (416) 593 - 5184

For more information contact CIPS or a member of the executive.

\*Includes Applicable G.S.T.



Those interested in advertising in the magazine, please write us to obtain a Press Kit.

Advertisers who reserve space for three consecutive issues are eligible for discounted rates.

## Advertising Notes

Arlene Merling, Managing Editor

*Canadian Artificial Intelligence*

Alberta Research Council

3rd Floor, 6815 - 8 Street N.E.

Calgary, Alberta, Canada

T2E 7H7 or — merling @arc.ab.ca

Telephone (403) 931-3390 Fax (403) 297-2339