

Diagnostic Algorithms Based on Multilevel Flow Models

Jan Eric Larsson

GoalArt, Tunavägen 39 C, 223 63 Lund, Sweden
Phone: +46 46 192630, E-mail: janeric@goalart.com

Abstract

Human error is a common source of accidents in complex plants. We believe that many human errors really are caused by lack of intelligence in the instrumentation and control systems, putting the operators in situations, which humans realistically cannot be expected to cope with. Through history, several computer-based algorithms have been proposed and used for automated sensor fault detection, alarm analysis, and fault diagnosis, to support human operators. The main problem with such algorithms is that they demand a large effort to build, validate, and especially rebuild when the plant is changed. We propose the use of algorithms based on Multilevel Flow Models (MFM), which are graphical models of *goals* and *functions* of technical systems. MFM provides a good basis for computer-based supervision and diagnosis, especially in real-time applications, where fast execution and guaranteed worst-case response times are essential. The expressive power of MFM is similar to that of rule-based expert systems, while the explicit representation of means-end knowledge and the graphical nature of the models make the knowledge engineering effort less and the execution efficiency higher than that of standard expert systems.

If MFM-based measurement validation and alarm analysis had been used, the Three-Mile Island incident would not have happened.

Introduction: Human Error

There are several different kinds of causes of accidents in large industrial plants. Many accidents are caused by failures in the physical hardware or the control system software, while others are caused by insufficient or erroneous operation routines, training, and regulations. Yet another type of accident is caused by *human error*, which is the kind of accident where the human operators did not manage the plant correctly, even though the hardware was functioning, and the routines and training were fine. In fact, human error is a fairly common cause of accidents.

Complex accidents often have several causes. For example, the infamous Three-Mile Island incident was caused by a malfunctioning valve (pilot-operated release

valve, PORV), which remained open although the instruments showed that it had been closed. Thus, there were causes in both physical hardware (the valve), and the control system software (the erroneous indication). What really turned this into a serious incident, though, was that the operators did not *understand* the situation quickly enough. During several hours, they did not check the measurements downstream from the open valve, which would have told them that the valve had not closed, and that the reactor was losing steam. Not until the next shift came on was the valve checked, and by then the core was almost uncovered. In the senate hearings, the failure to understand the situation and check whether the valve had indeed been closed was judged a *human error*, Lees (1983).

An implicit conclusion may seem to be that when human error is the cause of an accident, there is nothing wrong with the hardware or software. However, we strongly believe that many human errors are partly caused by shortcomings in the design of the control and presentation systems.

For example, in order to quickly find small problems, plants are equipped with a large number of alarms. But in a large accident, this may mean that too many alarms are activated, so that the operators cannot keep up with them, and the alarm system may become counter-productive or even useless. For example, in the Three-Mile Island incident, the printer queue for the alarms was some three hours behind schedule, and more than 100 audio alarms were active simultaneously, Lees (1983). Alarm showers may consist of several hundred alarms in less than a minute. When operators fail to act correctly under such circumstances, we consider it wrong to speak of human error, because no human would be able to handle the situation correctly.

Improving Instrumentation and Control Systems

Several methods for improving instrumentation and control systems have been proposed. Among these are:

- *Sensor fault detection*, based on local monitoring of each sensor or global comparison between multiple, partly redundant sensors. The latter could possibly

have helped the operators to have some suspicions about the PORV in the Three-Mile Island incident.

- *Alarm analysis*, that is, separation of alarms into primary and consequential ones, where the latter can be suppressed. It is believed that the number of alarms activated during the Three-Mile Island incident could have been reduced by many orders of magnitude by an alarm analysis algorithm.
- *Fault diagnosis*, where a computerized system performs measurements and asks questions in order to systematically find the primary explanations for a problem.
- *Failure mode and effects analysis*, where the consequences of breakdown of a certain physical component will be shown for other components and systems in the plant.

Alarm analysis systems were in use on the nuclear reactors at Oldbury and Wylfa in the United Kingdom, Lees (1983). These systems were based on alarm trees, that is, graphical descriptions where the possible alarms are linked to each other, telling which alarms are causally connected with each other.

However, these systems were not very successful. Referring to the Oldbury system, Long (1980) writes:

“However, the performance of this and two related systems was reported at the meeting to be less than satisfactory. Specifically, the alarm trees were *costly to develop*, subject to error, and *difficult to modify*.”

In later years, and especially after the Three-Mile Island incident, people have tried to use rule-based expert systems for automated fault diagnosis of complex plants. Again, the conclusions have been that the effort to build and update the knowledge needed in such a system is too large. Still, systems are constructed by, for example, Gensym Corporation and Cogsys, where the latter has built a system for alarm analysis based on fuzzy rules for a blast furnace plant in Australia.

In this paper, we present a set of algorithms for operator support, based on multilevel flow models. The main advantage is that the knowledge engineering effort needed is relatively small. Thus, we believe that these methods may indeed form a practical solution to many of the problems described, and help to avoid several kinds of human error.

Multilevel Flow Models

Multilevel flow models (MFM) are graphical models of *goals* and *functions* of technical systems. The goals describe the purposes of a system or subsystem, and the functions describe the capabilities of the system in terms of flows of *mass*, *energy*, and *information*. MFM also

describes the relations between the goals and the functions that *achieve* those goals, and between functions and the subgoals, which provide *conditions* for these functions. MFM was invented by Morten Lind at the Technical University of Denmark, see Lind (1990 a). Several new algorithms and implementations were contributed by Jan Eric Larsson at Lund Institute of Technology, see Larsson (1992, 1994 a, 1996).

MFM provides a good basis for diagnostic algorithms. The work of Larsson (1996) describes three algorithms based on MFM: measurement validation, alarm analysis, and fault diagnosis. Other algorithms have been developed later, such as fuzzy alarm analysis, see Dahlstrand (1998), Larsson and Dahlstrand (1998), failure mode analysis, see Öhman (1999), and sensor fault detection.

The measurement validation algorithm would have detected the discrepancy in the PORV flow at Three-mile Island, and the alarm analysis would have drastically reduced the number of active alarms. Had these MFM algorithms been in use, the incident would never have happened.

An Example of an MFM Model

MFM has been thoroughly explained in Lind (1990 a) and Larsson (1992, 1996). Here a small example will be given, to show the basics of MFM modeling. We will use a part of the main circulation system of a nuclear power plant. A much simplified process graph, from an example in the master's project Ingström (1998), is shown in Figure 1.

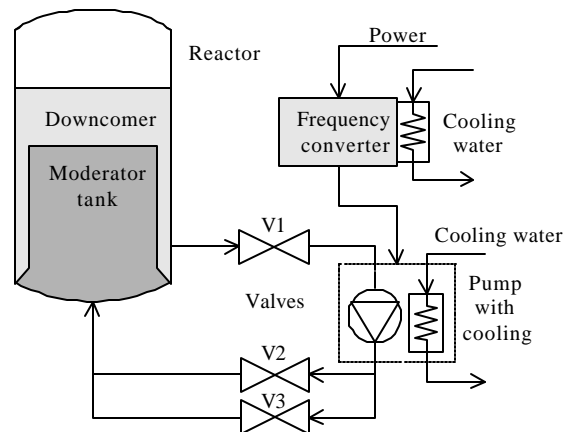


Figure 1. A process graph of the main recirculation system of a nuclear power plant.

In this system, reactor tank water flows from the downcomer, via the valve V1, to the pump. After the pump, the water flows through the two parallel valves

V2 and V3, back to the moderator tank. The pump is cooled by water. There is also a need for a frequency converter for the power to the pump, since the pump is frequency-controlled. Finally, the frequency converter must also be cooled. The purpose of the main water circulation is to control (moderate) the flow of neutrons in the reactor, and to cool it at the same time.

The goals of this simple system are: “maintain desired water flow through the moderator tank,” “cool the pump,” “provide electrical energy with the correct frequency,” and “cool the frequency transformer.”

The functions of the system are, among others, the downcomer’s ability to provide water, the pump’s ability to transport water, and the heat exchanger’s ability to transport heat. An MFM model of this system is shown in Figure 2.

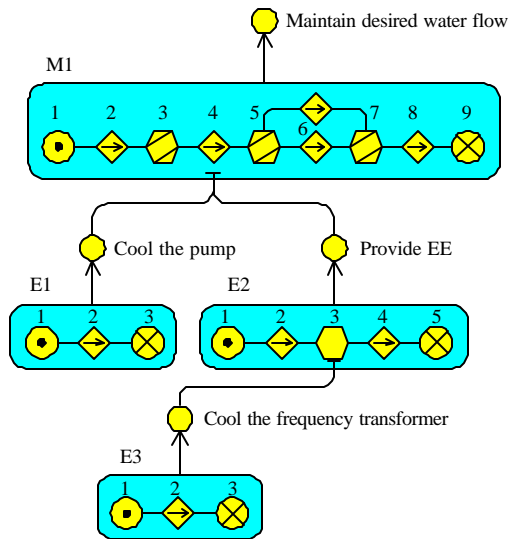


Figure 2. An MFM model of the main recirculation system.

In the MFM model, there are four flows. The flow network M1 describes the water flow from the downcomer to the moderator tank. The network E1 describes the transport of thermal energy from the pump to the cooling water. The network E2 describes the flow of electrical energy from the supply, via the frequency transformer, to the pump. Finally, the network E3 describes the flow of thermal energy from the frequency transformer to the cooling water. Thus, M1 is a model of a mass flow, and E1 to E3 are models of energy flows. In the network M1 the functions are, from left to right: 1) a source of water, realized by the downcomer; 2) a transport, realized by the valve V1; 3) a balance, realized by the pipe between V1 and the pump; 4) another transport, realized by the pump; 5) another

balance, realized by the forking pipe between the pump and the two parallel valves V2 and V3; 6) two transports, realized by the valves V2 and V3; 7) a balance, realized by the pipe sections between V2 and V3, and the moderator tank; 8) a transport, realized by the pipe that runs into the moderator tank; and finally, 9) a sink, realized by the moderator tank. The networks E1 to E3 contain energy flow functions describing the flows of electrical and thermal energy.

It should be noted that MFM describes how different flows enable each other. In the simple example in Figure 2, it can be seen that the cooling water flow E3 is necessary for the proper function of the frequency converter, and that the cooling water flow E1 and the electrical flow E2 are needed to keep the main water flow operating.

Algorithms Based on MFM

Over the years, Larsson and his research group have developed several algorithms based on MFM. The algorithms are as follows:

Quantitative Sensor Validation

This algorithm uses quantitative process measurements to detect inconsistencies between redundant sensor values. In this way, it can detect faulty sensors and leaks. It can also provide guesses about the correct values, which can be used as “validated” values instead of the faulty ones.

Discrete Sensor Validation

This algorithm uses discrete (alarm) values to detect inconsistencies between redundant indications. In this way, it can detect faulty sensors and leaks. It can also provide guesses about the correct values, which can be used as “validated” values instead of the faulty ones.

Alarm Analysis

This algorithm sorts discrete status indicators, such as events and alarms, into primary and consequential. In this way, it can pinpoint the root causes of large alarm showers correctly, and it allows for alarm suppression without risking suppression of the primary cause.

Fault Diagnosis

This algorithm uses discrete process measurements to search from observed fault indications to root causes. The result is a complete explanation of a fault situation. The algorithm uses fault observations to guide the search and avoiding unnecessary measurements.

Action Planning

This algorithm uses the results of the other algorithms to generate fault reports in different formats, including recommendations on corrective action plans.

Failure Mode Analysis

This algorithm calculates future consequences of actions, given a process state and one or several proposed faults or actions. In this way, it is an on-line planning support tool.

Reliability Analysis

This algorithm uses known measures of availability or reliability for each component to calculate similar measures for subsystems and entire processes, either off-line during design or on-line during operation.

Trend Analysis

This algorithm uses historical signal data (trend data) to provide early warning of faults, before alarm limits are exceeded.

Alarm Cleanup

This algorithm uses comparison of redundant status indications and alarms to detect erroneously set alarm limits, that is, to pinpoint alarms that are activated when they should not be, and alarms, which are silent when something is wrong.

Verification of Redundancy

This algorithm checks whether redundant subsystems rely on common support systems. If so, true redundancy may be compromised and the process design is faulty.

Verification of Safety Classification

This algorithm checks whether classified subsystems rely on non-classified support systems. If so, safety may be compromised

Verification of Redundancy

This algorithm checks whether redundant subsystems rely on common support systems. If so, true redundancy may be compromised.

All the algorithms above use the same MFM model, that is, the MFM model is the “knowledge database” for the algorithms. This has some obvious advantages:

- A single modeling effort will provide the database needed for a whole set of different diagnostic tasks.
- The same MFM model can be used throughout the life cycle of the process, for different design and supervision tasks.

Advantages of MFM Algorithms

The algorithms described in Larsson (1996) are based on discrete logic where the “sensor” values are *low*, *normal*, or *high*, and the resulting values are *consistent* or *inconsistent*, *working* or *failed*, *primary* or *consequential*, etc. In other words, MFM uses a linguistic interpretation of logic variables, just as do rule-based expert systems and systems based on fuzzy

logic. In addition, the MFM algorithms all operate by searching in fixed graphs. We have aimed at always producing algorithms that can handle the full MFM syntax, including closed loops in both the flows and the means-end dimension, as well as every kind of multiple fault situation. In addition, these complex cases should be handled by search methods of linear or sublinear complexity. So far, all of our presented methods fulfill these requirements. Together with the discrete logic, explicit means-end concepts, and graphical nature of MFM, this gives several advantages:

- The explicit description of goals and functions gives a small semantic gap between the diagnostic task formulation and the knowledge representation.
- The graphical representation provides strong support for knowledge base overview and consistency, and there is no need for a specialized knowledge engineering tool.
- The high level of abstraction makes knowledge acquisition, knowledge engineering, and knowledge base validation and support considerably easier than with standard rule-based systems or fuzzy logic systems.
- The graphical nature of the models allows the algorithms to have good real-time properties, such as an easily computed worst-case time, low memory demands, and high efficiency.
- The high level of abstraction allows the algorithms to be very fast. A worst-case fault diagnosis on the Guardian system, for example, takes less than 80 microseconds on a 500 MHz Pentium Computer.

These advantages have been observed in practice, during the development of the Steritherm system and during the Guardian project, when MFM was compared to several other modeling methodologies, see Larsson (1996), Larsson et al. (1997 b), and Larsson and Hayes-Roth (1998). Furthermore, we have corroborated these evaluations in an alarm analysis project for the Barsebäck nuclear power plant, Larsson (1998), Larsson and Öhman (1998), Öhman (2000 a, b).

Experiences of MFM from the Steritherm Project

The author’s doctor’s project used two target processes, a small lab tanks system and Steritherm, Larsson (1992). The latter is a widely used, moderately sized process for ultra-high temperature (UHT) treatment of dairy products.

The most important observation from this effort was that the knowledge engineering effort needed to build the MFM model of Steritherm was considerably less than for the other diagnostic methods also used in the KBRTCS project. These other algorithms were MIDAS, Oyeleye (1989) and Finch (1989), a system using signed

directed graphs, (SDG), and the Diagnostic Model Processor, (DMP), Petti et al. (1990), Petti and Dhurjati (1991), and Petti (1992), a representation based on quantitative equations.

Experiences of MFM from the Guardian Project

The Guardian project aimed at developing a monitoring and diagnosis system for use with post-operative intensive-care patients, see Larsson et al. (1997 a, b) and Larsson and Hayes-Roth (1998), and resulted in a demonstrator system which was successfully tested on realistic scenarios. In the verification tests that were performed during the project, the system outperformed the human test subjects, see Larsson et al. (1997 b).

For the Guardian project, a large MFM model of the human body was developed. It covers all systems needed for intensive-care unit monitoring. The algorithms provided accurate, reliable, and easily tuned diagnostics, and they were much faster than the other algorithms in Guardian. In addition, the knowledge engineering effort needed for the MFM model was clearly less than what was needed for the other methodologies.

The two other methods used in Guardian were REACT and PCT, (parsimonious covering theory), Larsson, Hayes-Roth, and Gaba (1997 a), Larsson, Hayes-Roth, Gaba, and Smith (1997 b), Larsson and Hayes-Roth (1998). Both these representations needed considerably more work than MFM, mainly because they rely on numerical weights for conditional probabilities for a sign to be observed given that a disease is present.

The Barsebäck Project

In this project, we developed MFM models of selected main systems of the Barsebäck nuclear power plant, in cooperation between the Department of Information Technology and Southern Sweden Power Supply (Sydkraft AB). The aim was to provide fast and reliable alarm analysis based on MFM. The master's thesis Ingström (1998) presented a first MFM model of the main systems of the power plant. Test scenarios came from the nuclear simulator facility at KSU, Studsvik. A demonstrator system was ready in the autumn of 2000 and the results so far have been described in Öhman (2000 a, b).

Related Work

The main contributions to MFM have been made by Morten Lind and his group. Lind (1990 a, 1994) describes the basics of MFM, while Lind (1990 b) contains an early suggestion for a diagnostic system. Lind has also treated real-time diagnosis, Lind (1990 c), and design of operator interfaces, Lind (1989).

MFM has also been used in nuclear safety research, De et al. (1982) and Businaro et al. (1985), in operator interfaces for fault diagnosis, Duncan and Prætorius (1989), for constructing COGSYS diagnostic systems, Sassen (1993), for fault diagnosis in process industry, Walseth (1993), and in intelligent man-machine systems for nuclear plants, Monta et al. (1991).

MFM can be compared to other modeling and diagnosis methodologies, such as rule-based expert systems, fuzzy logic, qualitative physics based on Reiter's algorithm, Hamscher et al. (1992), Reiter (1987), Greiner et al. (1989), classical statistical methods, methods from control theory, Frank (1996), and neural networks. In comparison to expert systems and fuzzy logic, MFM imposes a *deep* model structure of means and ends, as opposed to a *shallow* rule-based representation. It differs from qualitative physics in that it explicitly represents goals and functions, avoids general logic, and is computationally more efficient, while qualitative physics has been geared towards diagnosis of electrical circuits, a task which MFM is not very well adapted for. MFM differs from statistical and control theory methods in that it uses discrete and more abstract representations, and thus is useful on a higher level of decision and diagnosis. For example, control theory methods are usually aimed at *fault detection* on control loop level, while MFM is aimed at diagnostic reasoning on a plant-wide level. Finally, MFM differs strongly from neural networks in that it explicitly represents human knowledge using linguistic concepts, and that the model construction relies almost completely on available human knowledge and not on automatic generalization of test cases.

Conclusions

MFM provides a good basis for diagnostic algorithms for industrial processes. Among its advantages is an explicit description of goals and functions, a relatively easy knowledge engineering task due to the graphical and highly abstract nature of MFM models, and finally, the possibility to produce very fast algorithms with good real-time properties. With MFM, it is possible to reduce the number of accidents caused by human error.

Acknowledgements

The author would like to thank Fredrik Dahlstrand and Bengt Öhman at the Department of Information Technology for their excellent and inspiring efforts. Great thanks go to Morten Lind, who invented MFM and have supported our efforts from the beginning. Finally, I would like to thank Anu Uus for giving several suggestions on how to improve this paper.

The paper is based in part on Larsson (2000, 2002).

References

- Businaro, T., A. Di Lorenzo, G. B. Meo, M. I. Rabbani, and E. Rubino, "An Application of MFM Method for Nuclear Plant State Identification," Proceedings of the Halden Programmer's Group Meeting on Computerized Man-Machine Communication, Göteborg, 1985.
- Dahlstrand, F., "Alarm Analysis with Fuzzy Logic and Multilevel Flow Models," Proceedings of the 18th Annual International Conference of the British Computer Society Special Group on Expert Systems, ES98, Cambridge, England, pp.173-188, 1998.
- De, M. K., J. A. Rumancik, A. J. Impink, and J. R. Easter, "A Functional Design Approach to PWR Safety," Proceedings of the International Meeting on Thermal Nuclear Reactor Safety, Chicago, Illinois, 1982.
- Duncan, K. D. and N. Prætorius, "Flow Displays Representing Complex Plant for Diagnosis and Process Control," Proceedings of the 2nd European Meeting on Cognitive Science Approaches to Process Control, Siena, 1989.
- Finch, F. E., *Automated Fault Diagnosis of Chemical Process Plants Using Model-Based Reasoning*, Doctor's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1989.
- Frank, P. M., "Analytical and Qualitative Model-Based Fault Diagnosis ? A Survey and Some New Results," *European Journal of Control*, vol. 2, pp. 6-28, 1996.
- Greiner, R., B. A. Smith, and R. W. Wilkerson, "A Correction to the Algorithm in Reiter's Theory of Diagnosis," *Artificial Intelligence*, vol. 41, pp. 79-88, 1989.
- Hamscher, W., L. Console, and J. de Kleer, (Eds.), *Readings in Model-Based Diagnosis*, Morgan Kaufmann, San Mateo, California, 1992.
- Ingström, D., "MFM Modeling and Alarm Analysis of the Barsebäck Nuclear Power Plant," Master's thesis, Department of Information Technology, Lund Institute of Technology, Lund, 1998.
- Larsson, J. E., *Knowledge-Based Methods for Control Systems*, Doctor's thesis, TFRT-1040, Department of Automatic Control, Lund Institute of Technology, Lund, 1992.
- Larsson, J. E., "Diagnostic Reasoning Strategies for Means-End Models," *Automatica*, vol. 30, no. 5, pp. 775-787, 1994 a.
- Larsson, J. E., "Hyperfast Algorithms for Model-Based Diagnosis," Proceedings of the IEEE/IFAC Joint Symposium on Computer-Aided Control Systems Design, Tucson, Arizona, 1994 b.
- Larsson, J. E., "Diagnosis Based on Explicit Means-End Models," *Artificial Intelligence*, vol. 80, no. 1, pp. 29-93, 1996.
- Larsson, J. E., "Alarm Analysis for a Nuclear Power Plant Using Multilevel Flow Models," invited paper, Proceedings of the 9th International Symposium on System, Modeling, Control, Zakopane, Poland, 1998.
- Larsson, J. E., "Avoiding Human Error," Proceedings of the International Conference on Control and Instrumentation in Nuclear Installations, Bristol, England, 2000.
- Larsson, J. E., "Diagnostic Reasoning Based on Means-End Models: Experiences and Future Prospects," *Knowledge-Based Systems*, vol. 15, no. 1-2, pp. 103-110, 2002.
- Larsson, J. E. and F. Dahlstrand, "New Algorithms for MFM Alarm Analysis," invited paper, Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, San Diego, California, 1998.
- Larsson, J. E. and B. Hayes-Roth, "Guardian: An Intelligent Autonomous Agent for Medical Monitoring and Diagnosis," *IEEE Intelligent Systems*, vol. 13, no. 1, pp. 58-64, 1998.
- Larsson, J. E., B. Hayes-Roth, and D. M. Gaba, "Goals and Functions of the Human Body: An MFM Model for Fault Diagnosis," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 27, no. 6, pp. 758-765, 1997 a.
- Larsson, J. E., B. Hayes-Roth, D. M. Gaba, and B. E. Smith, "Evaluation of a Medical Diagnosis System Using Simulator Test Scenarios," *Artificial Intelligence in Medicine*, vol. 11, pp. 119-140, 1997 b.
- Larsson, J. E. and B. Öhman, "Model-Based Alarm Analysis for Large Plants," invited paper, Proceedings of the International Conference on Systems, Signals, Control, Computers, Durban, South Africa, 1998.
- Lees, F. P., "Process Computer Alarm and Disturbance Analysis: Review of the State of the Art," *Computer and Chemical Engineering*, vol. 7, no. 6, pp. 669-694, 1983.
- Lind, M., "Human-Machine Interface for Diagnosis Based on Multilevel Flow Modeling," Proceedings of the 2nd European Meeting on Cognitive Science Approaches to Process Control, Siena, 1989.
- Lind, M., "Representing Goals and Functions of Complex Systems — An Introduction to Multilevel Flow Modeling," Technical report, 90-D-38, Institute of Automatic Control Systems, Technical University of Denmark, Lyngby, 1990 a.
- Lind, M., "Abstractions Version 1.0 — Descriptions of Classes and Their Use," Technical report, 90-D-380, Institute of Automatic Control Systems, Technical University of Denmark, Lyngby, 1990 b.
- Lind, M., "An Architecture for Real-Time MFM Diagnosis," Technical report, Institute of Automatic Control Systems, Technical University of Denmark, Lyngby, 1990 c.
- Lind, M., "Modeling Goals and Functions of Complex Industrial Plants," *Applied Artificial Intelligence*, vol. 8, no. 2, pp. 259-283, 1994.
- Long, A. B., "Technical Assessment of Disturbance Analysis Systems," *Nuclear Safety*, vol. 21, no. 38, 1980.
- Monta, K., J. Takizawa, Y. Hattori, T. Hayashi, N. Sato, J. Itoh, A. Sakuma, and E. Yoshikawa, "An Intelligent Man-Machine System for BWR Nuclear Power Plants," Proceedings of AI91 — Frontiers in Innovative Computing for the Nuclear Industry, Jackson, Wyoming, 1991.
- Öhman, B., "Failure Mode Analysis Using Multilevel Flow Models," Proceedings of the 5th European Control Conference, Karlsruhe, Germany, 1999.
- Öhman, B., "Code Generation for Alarm Analysis with Multilevel Flow Models," Proceedings of the Second International Symposium on Engineering of Intelligent Systems, EIS '2000, Paisley, Scotland, 2000 a.
- Öhman, B., "Alarm Analysis on Large Systems Using Multilevel Flow Models," Proceedings of the IFAC Symposium on Artificial Intelligence in Real-Time Control, Budapest, AIRTC-2000, Hungary, 2000 b, to appear.
- Oyleye, O. O., *Qualitative Modeling of Continuous Chemical Processes and Applications to Fault Diagnosis*, Doctor's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1989.
- Petti, T. F., *Using Mathematical Models in Knowledge-Based Control Systems*, Doctor's thesis, University of Delaware, Newark, Delaware, 1992.
- Petti, T. F. and P. S. Dhurjati, "Object-Based Automated Fault Diagnosis," *Chemical Engineering Communications*, vol. 102, pp. 107-126, 1991.
- Petti, T. F., J. Klein, and P. S. Dhurjati, "Diagnostic Model Processor: Using Deep Knowledge for Process Fault Diagnosis," *AIChE Journal*, vol. 36, no. 4, pp. 565-575, 1990.
- Reiter, R., "A Theory of Diagnosis from First Principles," *Artificial Intelligence*, vol. 32, pp. 732-737, 1987.
- Sassen, J. M. A., *Design Issues of Human Operator Support Systems*, Doctor's thesis, Faculty of Mechanical Engineering and Marine Technology, Laboratory for Measurement and Control, Delft University of Technology, Delft, 1993.
- Walseth, J. Å., *Diagnostic Reasoning in Continuous Systems*, Doctor's thesis, ITK-rapport 1993: 164-W, Division of Engineering Cybernetics, Norwegian Institute of Technology, Trondheim, 1993.