KNOWLEDGE ENGINEERING USING MULTILEVEL FLOW MODELS

Jan Eric Larsson
Department of Information Technology
Lund Institute of Technology
Box 118, 221 00 Lund, Sweden
E-mail: janeric@it.lth.se

Abstract – Multilevel flow models (MFM) describe goals and functions of a technical system. The goals are the objectives of running the system, and the functions are its capabilities in terms of mass and energy flows. Thus, MFM describes the intentionality of the target system in linguistic terms. The resulting models can be used for different diagnostic tasks, such as fault diagnosis, causal explanations, and qualitative predictions. MFM has several properties which makes for a relatively easy *knowledge engineering* task, compared to mathematical models as used in classical control theory and compared to the rule bases used in standard expert systems. In addition, MFM allows for diagnostic algorithms with excellent real-time properties.

I. INTRODUCTION

Multilevel Flow Models (MFM) are graphical models of *goals* and *functions* of technical processes. The goals describe the purposes of a system and its subsystems, and the functions describe the system's abilities 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).

Several diagnostic algorithms have been developed with MFM as a basis. The work of Larsson (1992, 1994 a, 1996) describes algorithms for checking consistency between redundant sensor values, for separating primary and consequential faults in multiple fault situations, fault diagnosis, and generation of explanations in pseudo-natural language. Öhman (1999) presents an algorithm for failure mode analysis. Dahlstrand (1998) presents an algorithm based on fuzzy logic for separating primary and consequential alarms.

This paper treats *knowledge engineering for MFM*. While knowledge engineering in general is a difficult and time-consuming task, MFM has some properties, which makes for a relatively easier task than usual. The MFM semantics use concepts that are very abstract, high-level, and also, we believe, close to those used by human designers and operators. Our experiences are based on three projects, reported in Chapter IV, but first, a short description of MFM is given.

II. MULTILEVEL FLOW MODELS

MFM is a graphically represented, formal modeling language, in which the intentional properties of a technical system are described. The purposes of the system and its subsystems are modeled with goals, which can be either production goals, safety goals, or economy (or optimization) goals. The abilities of the systems are modeled with flow functions, connected into flow paths or functional networks. The main functions are sources, transports, storages, balances, barriers, and sinks, and they describe either mass or energy flows. Observers, decision makers, and actors describe information flows. The manager function describes control systems. Each flow network can be connected to one or several goals via *achieve relations*, which means that the functions in the network achieve the fulfillment of the goal. A goal may be connected to one or several functions via *condition relations*, meaning that the goal is a condition for the function. For a full description of MFM, see Larsson (1992 c, 1994 a, 1996) or Lind (1990 a). The symbols of the MFM graphical language are shown in Figure 1.

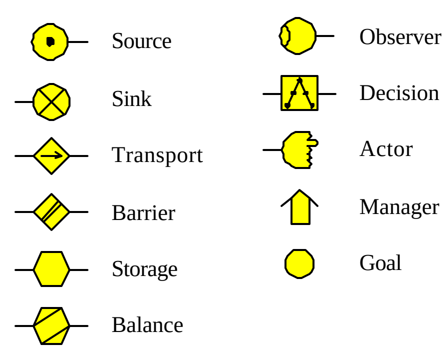

Figure 1. The symbols for different MFM objects.

III. AN EXAMPLE OF AN MFM MODEL

The nature of MFM models is probably best explained by an example. 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 2.

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.

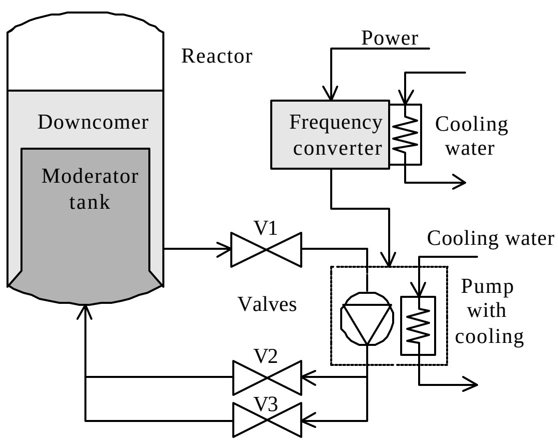

Figure 2. A process graph of the main recirculation system of a nuclear power plant.

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 3.

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.

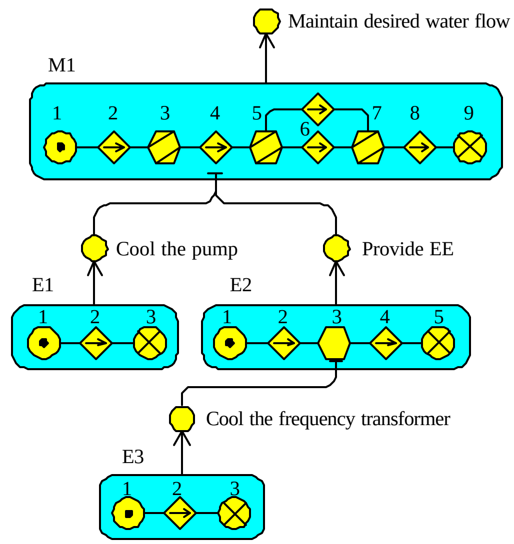

Figure 3. An MFM model of the main recirculation system.

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 3, 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. To our knowledge, the control systems used in today's power plants do not incorporate information on such dependencies.

IV. SOME PROPERTIES OF MFM

MFM has a well-defined syntax for how the different symbols may be connected, see Lind (1990 a) and Larsson (1992). There are a few minor differences between the syntax of Lind and Larsson, but all in all, the syntax concerning mass and energy flows, achieve relations, and conditions has reached a mature state. The semantic interpretation of MFM is still developing, however. Again, there are clear conventions as for how to model mass and energy flows, and Lind (1990 a) gives guidelines for this. The level of detail and the "style" of MFM models may vary, though. One of the hoped for results of our new projects (described further below) is to gain more experiences in MFM modeling "style" and practice.

Currently, MFM has only been applied to flow systems, where the flows are of mass, energy, and sometimes information. Although this means that MFM can describe a large class of target processes, there is still a possibility of proposing additions to MFM for describing chemical reactions, biological processes, geometrical and navigational concepts, logistics and planning, etc. For some analysis and suggestions on how to extend MFM, see Larsson (1992).

IV. KNOWLEDGE ENGINEERING

All the algorithms described in Larsson (1992) are based on discrete, qualitative 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 addition, the algorithms operate by searching in fixed graphs and can handle closed loops in both the flow networks and the means-end dimension, as well as multiple fault situations. These difficult cases are handled by search methods of linear or sublinear complexity. 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 means-end structure of MFM is particularly well-suited for designing models top down, which means that it is possible to describe large and complex processes in simple, plant-wide concepts and then to develop selected parts of the process in greater detail.
- 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 (see below), for example, takes less than 200 microseconds on a 200 MHz Pentium Pro Computer.

These advantages have been observed in practice, during the development of the Steritherm system, 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), and in a new alarm analysis project for the Barsebäck nuclear power plant, Larsson (1998), Larsson and Öhman (1998). These are the projects from which we have gained the experiences that will be described in the rest of the paper.

Building MFM Models

From our experiences of Steritherm, Guardian, and the ongoing nuclear power plant project, it is clear that MFM allows for a relatively small modeling effort. Work must go into several different tasks:

- *Learning how the process works*. This part of a knowledge engineering task demands a large effort no matter what modeling methodology is used, and it has been the most demanding effort in all three of our projects.
- *Deciding on the particular tasks of the system* and in what operational state these tasks should be fulfilled. Again, this is fairly independent of the modeling methodology, and it has meant quite an effort, especially in the nuclear power plant project.
- *Enforcing a structure of the model* that enables a systematic procedure of model building. Here, MFM's intentional top-down structure has clear advantages over the mathematical models used in control, the rule bases of standard expert systems, as well as the more detailed, bottom-up models of qualitative physics.
- *Development of a knowledge-engineering tool,* which matches the target area and the model structure. Here, the graphical nature of MFM and the

MFM Toolbox editor already provide an excellent tool.

- *Construction of the model*. Here, the high abstraction level of MFM makes for a much easier task than either mathematical equations or expert system rules.
- *Evaluation of the model*. Again, this task is fairly independent of the modeling methodology chosen, but it has meant reasonable efforts only in all three cases.
- *Inclusion of the resulting knowledge-based system in the supervision and control system*. This will mean severe problems for most knowledge-based approaches, where programming language, software system structure, and real-time demands often mean that the knowledge-based system must be run on a separate computer outside the conventional system. However, the MFM Toolbox generates standard C code and guarantees excellent real-time and memory-handling properties.

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. It is a real process in worldwide use, but still small enough to be of manageable size for an academic research project. It was the target process used in the project "Knowledge-Based Real-Time Control Systems" (KBRTCS), of which the author's doctor's project was a part, see Årzén (1993). The MFM model of Steritherm describes the toplevel goal of sterilizing the liquid foodstuff by heating it to 137 ºC for a few seconds. The main flow of thermal energy is modeled in detail, and the supporting flows of product, circulation water, and cooling are described. However, some other support systems, such as pressurized air and the 220 and 380 Volt electrical systems were not included in the model.

Algorithms for consistency between measurements, alarm analysis, and fault diagnosis were tested on Steritherm, under realistic conditions, with the conclusion that they gave correct and useful information. They all handled multiple faults without problems. The most important observation 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. The knowledge engineering needed for MIDAS was clearly larger than for MFM, partly because SDG have less inherent structure than MFM, and partly because MIDAS needed rather much numerical parameter tuning at the lower-level input layer. DMP demanded a large knowledge engineering effort mainly because it needs quantitative equations to describe the system.

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 limited number of verification tests that were performed during the project, the system outperformed the human test subjects, see Larsson et al. (1997 b). The alarm analysis and fault diagnosis algorithms were implemented in Common Lisp and integrated into the Guardian system architecture. There was also an algorithm for generating a standard backward-chaining rule base for fault diagnosis from an MFM model.

For the Guardian project, a large MFM model of the human body was developed. It covers all systems needed for intensive-care unit monitoring, such as the heart, circulation, the body fluid volume, the nutrition, respiration, oxygen and carbon dioxide concentrations, body temperature, acid-base balance, the concentrations of sodium and potassium, and the regulatory mechanisms for these systems. The model in its final version consists of some 500 MFM objects and corresponds to a rule base of some 400-800 rules, that is, a knowledge-based system of reasonable size. The use of MFM in Guardian was very successful. 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. Such conditional probabilities are either not known or not well documented, and since the different diseases are not independent, the statistical assumptions of PCT are violated, and the parameters need to be hand tuned for the method to work well, Larsson, Hayes-Roth, Gaba, and Smith (1997 b). In addition, the MFM algorithm turned out to be orders of magnitude faster than the other algorithms, even though REACT was designed especially for fast reactive diagnosis.

V. REAL-TIME MFM ALGORITHMS

In addition to allowing an efficient knowledge engineering, the MFM algorithms themselves are efficient and have good real-time properties. They are based on searches in fixed graphs, which makes it possible to calculate a worst-case execution time. In practice, this is easily done by running a thousand or so diagnoses and timing them. The worst case for a fault diagnosis on the entire Guardian model was 1100 microseconds on a SUN SPARC station 2, which is much faster than most knowledge-based systems, see Larsson (1994 b). The speed of the MFM algorithms allows us to use a large and complex system like a nuclear power plant as target process in an ongoing project, Larsson and Öhman (1998).

VI. 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.

MFM share some properties with each of these other methodologies, while other properties are complementary. Thus, a realistic system for supervision and diagnosis based on MFM will also have to contain a selection of other models and algorithms, for handling problems where the other method may be better suited than an MFM algorithm. The architecture of such systems has been hinted at in Larsson (1992).

VII. CONCLUSIONS

MFM provides a good basis for diagnostic algorithms for industrial processes. Among its advantages are 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.

VIII. 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.

IX. REFERENCES

Årzén, K.-E., "Using Multi-View Objects for Structuring Plant Databases," *Intelligent Systems Engineering*, vol. 2, no. 3, pp. 183–200, 1993.

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. 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.

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.

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.

Oyeleye, 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.