

# Diagnostic Reasoning Based on Means-End Models: Experiences and Future Prospects

Jan Eric Larsson

Department of Information Technology, Lund Institute of Technology, Box 118, 221 00 Lund, Sweden  
Phone: +46 46 222 7523, Fax: +46 46 222 4714, E-mail: janeic@it.lth.se

## Abstract

Multilevel Flow Models (MFM) are graphical models of *goals* and *functions* of technical systems. MFM was invented by Morten Lind at the Technical University of Denmark and several new algorithms and implementations have been contributed by the group headed by Jan Eric Larsson at Lund Institute of Technology. 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. 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. The paper gives an overview of existing MFM algorithms, and different MFM projects which have been performed or are currently in progress.

## Introduction

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 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 four algorithms based on MFM. *Measurement validation* checks consistency between redundant sensor values, and can discover flow leaks, sensor failures, and other measurement errors. The *alarm analysis* algorithm analyses any (multiple) fault situation and can tell which

faults are primary and which faults that may be consequences of the primary ones. The *fault diagnosis* uses sensor values and queries to the operator to discover the faults of the target system. The *explanation generation* algorithm uses the states discovered by the fault diagnosis to produce explanations and remedies in pseudo-natural language. Other algorithms have been developed later. The *failure mode analysis* uses MFM with added timing information to predict the consequences of failures. It can be used both during the design phase of a plant and in real-time during actual operation, Öhman (2000 a, b). The *fuzzy alarm analysis* works in a way similar to the discrete alarm analysis, but is based on fuzzy logic, which makes it more robust when faced with noisy signals close to decision boundaries, see Dahlstrand (1998), Larsson and Dahlstrand (1998).

MFM research is currently in a phase of maturing and further development, and this paper aims at giving an overview of both what has been done and what it is currently ongoing. It is focused mainly on the efforts of the author's research group at the Department of Information Technology, Lund Institute of Technology, Sweden. This paper also 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.

## 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 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.

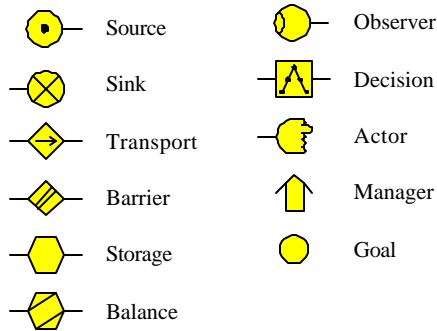


Figure 1. The symbols for different MFM objects.

**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.

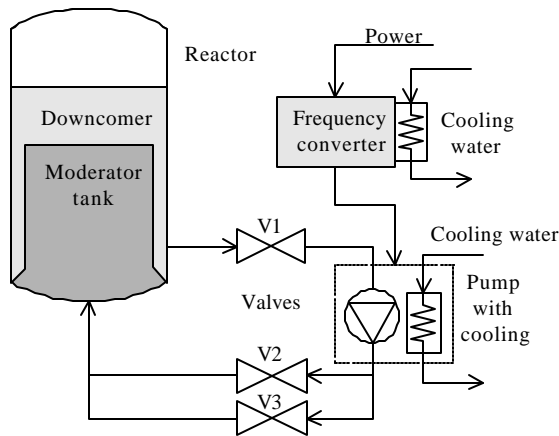


Figure 2. A process graph of the main recirculation system of a nuclear power plant.

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 3.

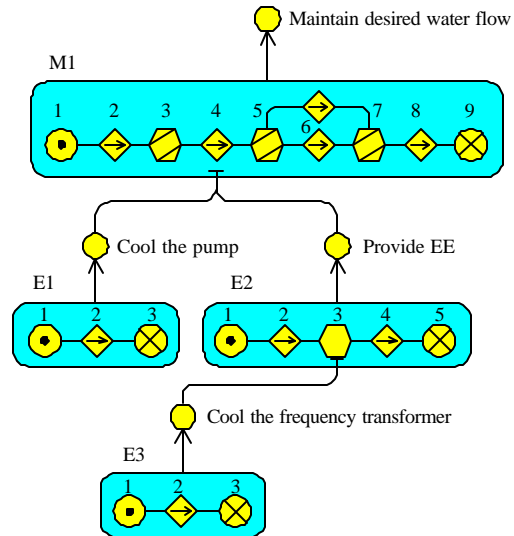


Figure 3. An MFM model of the main recirculation system.

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 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.

### Advantages of MFM Algorithms

All 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 Pro Computer.

These advantages have been observed in practise, 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 corroborated these evaluations in the alarm analysis project for the Barsebäck nuclear

power plant, Larsson (1998), Larsson and Öhman (1998), Öhman (2000 a, b).

### 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.

### Projects at Automatic Control, Lund University

During the author's doctor's project, three new algorithms based on MFM were invented and implemented in the expert system shell G2. The algorithms were *measurement validation*, *alarm analysis*, and *fault diagnosis*, see Larsson (1992, 1994 a, 1996). The implementation resulted in an MFM Toolbox in G2, which was offered as a product. It has been used as part of

several doctor's projects at Morten Lind's department at the Technical University of Denmark and sold to CERN.

The author's doctor's project used two target processes, the tanks system and Steritherm. The tanks process is a small laboratory process used in teaching basic control theory at the Department of Automatic Control, Lund Institute of Technology, Sweden. It was used as the generic "toy" example during the development of the algorithms, and has remained a standard example ever since.

Steritherm 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 academic research projects. 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 °C 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.

All three algorithms 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.

### **The Guardian Project, Stanford University**

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. A fourth algorithm was also invented, the *explanation generation*, which provides explanations of fault situations in pseudo-natural language. 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 spite of the fact that one of these other algorithms was designed specifically for speed.

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.

### **Hyperfast Model-Based Diagnosis**

At the end of the author's doctor's project, when the G2 implementation was finished and tested, a small test implementation was also made in standard C for a SUN SPARC station. This implementation proved to be very simple, compact, and efficient. Therefore, during the Guardian project, the author implemented a new version of all the algorithms in C, together with a graphical editor and a file system, the latter two for the Macintosh. The result of this effort was the MFM Toolbox Version 1.0, which was used as the model building tool for the Guardian project. In the summer of 1997, the toolbox was

reimplemented in C++ for Windows '95 and NT, to become the MFM Toolbox Version 2.0. This version performs more than 12 500 diagnoses per second on the Guardian model.

The MFM Toolbox demonstrated that the MFM algorithms are very efficient and have good real-time properties. First, the algorithms are based on searches in fixed graphs, which makes it possible to calculate a worst-case time. In practise, this is easily done by running a thousand or so diagnoses and time them. The worst case for a fault diagnosis on the entire Guardian model is 12000 complete diagnoses per second (corresponds to an execution speed of 5 million rules per second), 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), Öhman (2000 a, b).

#### **Alarm Analysis for a Nuclear Power Plant**

Monitoring and control of modern large-scale industrial processes is often difficult, due to the complexity of the processes. Therefore, efficient methods for alarm analysis and structuring of control systems are needed. In a complex fault situation, many alarms may trigger at the same time, where some of the alarms are due to primary faults, and others may only be symptoms of consequential faults. In a current project, we are using the alarm analysis method to provide operators with a decision support tool to use in complex fault situations.

In this project, we are developing an MFM model of the 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 is 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, and gave a foundation for a larger model. One important experience gained from this master's project was that it is difficult to find information on large and complicated alarm situations pertaining to normal operation. Therefore, an MFM model of the power plant's normal operating state may be inadequate. Instead, it seems probable that large and complex fault situations will usually occur in already failed states, when the main objective of the operators is to shut down the plant. In order to find out in what situations there is a need for complex alarm analysis, we are cooperating with the nuclear simulator facility at KSU, Studsvik. At Studsvik, seven different simulators have been constructed (one for each type of control room currently in use in Swedish nuclear power plants), and these are used to train operators in handling fault situations. We have analyzed several operator training scenarios and used the results to decide for which operating states we should produce and test MFM models.

#### **New Algorithms Based on Fuzzy Logic**

In another project, the goal has been to develop and evaluate new versions of the MFM algorithms based on a better support from underlying data validation algorithms, and by using fuzzy logic in the algorithms themselves, Dahlstrand (1998, 2000 a, b, c). In this way, we could make the current discrete algorithms better adapted to problems with noise, missing data, etc. The crisp logic MFM alarm analysis algorithm may have problems when faced with noisy signals, and signals close to decision limits. Fuzzy logic is an interesting approach to come to terms with some of the problems concerning the uncertainties in the real world. The following benefits are gained when combining the existing MFM alarm analysis with fuzzy logic:

- It is possible to grade the closeness to a decision limit.
- It is possible to grade a failure, that is, "how failed" a function is.
- The alarm analysis algorithm is more reliable when there are disturbances caused by small and rapid changes.
- The rules and the algorithm of the already existing MFM alarm analysis algorithm may be used.

MFM alarm analysis using fuzzy logic solves some of the problems the already existing discrete logic algorithm would not handle, for example, chaotic switching due to noise and closeness to a decision limit. One possible drawback of the fuzzy alarm analysis algorithm is that it is slower than the already existing algorithm, because of the increased computational complexity. However, since the already existing MFM algorithm is very fast, the fuzzy logic algorithm is still among the fastest algorithms available.

#### **Adviser: A Next Generation of Guardian**

It is our plan to start another project in which MFM would play an essential role. The aim of this project would be to build a next generation of Guardian, based on C/C++ and Windows NT instead of Lisp, and smaller, faster, and more reliable, so that the emphasis of the project can be knowledge acquisition, testing, and verification. So far, one very successful master's project called *Adviser* has been performed, see Bengtsson and Bäckwall (1998). In this project, a demonstration program was created, which gives a good view of how a system used in real situations could look. It is a small and fast real time system with an easy-to-use user interface. The first version of the system is written in Java and runs on a standard PC. It contains a simple data generator, a rule-based expert system using backward chaining, and the user interface. We plan to use this demonstrator system as a starting point for a Ph.D.-level project involving both medical and computer science students.

Within the Adviser project, we also investigated the more specific problem of monitoring and diagnosis of the heart. The two efforts were to design a knowledge-based toolbox for ECG analysis and to build a detailed MFM model of the heart's physical and electrical (nervous) functions. The ECG analysis toolbox contains an ECG signal generator (a simulator), feature detection based on classical methods from statistics, and fault diagnosis based on discrete rule-based systems, and fuzzy logic, Dubowik (1999). We plan to make this toolbox an integral part of the Adviser system, and the heart model will allow us to test our MFM algorithms on realistic data from an interesting domain, which is very different in nature from that of nuclear power plants and other technical systems.

### **An Algorithm for Failure Mode Analysis**

In the design of a complex industrial system, it is important to study the effects of failures on other parts of the system. Traditionally, this has been done using manual methods, by filling out forms by hand, such as the Failure Mode and Effects Analysis method (FMEA). There are methods with automated computer support, such as Fault Tree Analysis, and tools are available to automate parts of an FMEA analysis, see Price et al. (1997). There is now a new algorithm for failure mode analysis based on the consequence propagation used in the alarm analysis algorithm, expanded with timing information associated with condition relations and storages, see Öhman (1998). The method takes a given target system state as input, and outputs a list of predicted time-to-failure values for the affected parts of the system. A shortcoming with the common methods for failure mode analysis is that the analysis is done during the design phase, and in practise, the results of the analysis are not easily retrievable by the operators during operation. It may also be the case that some of the failure modes have not been taken into consideration during the previous analysis. The FMA algorithm in MFM is capable of both off-line and on-line analysis, and could be used in both the design phase for a system, and when the system is running.

### **The MFM Toolbox Version 3.0**

Alongside the other ongoing projects, we are also developing a new version of the MFM Toolbox in an object-oriented fashion, using C++ under Windows NT. This toolbox will contain an intelligent graphical editor, the algorithms themselves, and a possibility of producing small versions of the algorithms for embedded systems. These small MFM systems will be generated either by downloading data files into an MFM micro kernel, or by using a code generator, which produces C/C++ subroutines with algorithms and hard-coded MFM models in them.

### **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. These efforts of Lind's group used Smalltalk 80 as the main tool for implementation. Lind has also treated real-time diagnosis, Lind (1990 c), and design of operator interfaces, Lind (1989). Lind's group has developed a graphical interface, Duschek (1991) and Osman (1992), a STRIPS planning system, Larsen (1993), and a fault diagnosis system for ship engines, Jørgensen (1993). More lately, Lind's group has presented a combination of MFM and the Goal Tree-Success Tree (GTST) approach of Modarres, see Jalashgar (1997). GTST has been used for fault diagnosis, Chung and Modarres (1989), alarm analysis, Modarres and Cadman (1986), and operator support, Kim and Modarres (1987).

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 were the other method may be better suited than an MFM algorithm. The architecture of such systems has been hinted at in Larsson (1992).

### 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. Research within the MFM area is maturing, and several new projects are ongoing. Hopefully, methods based on MFM will be brought into industrial practise within the next 10 years.

### Acknowledgments

The author would like to thank Fredrik Dahlstrand and Bengt Öhman at the Department of Information Technology for their excellent and inspiring efforts. Also, Jonas Bengtsson, Lars-Göran Bäckwall, Daniel Ingström, and Thomas Mårtensson are acknowledged for their excellent master's projects. I would also like to thank Karl Johan Åström and Karl-Erik Årzén at the Department of Automatic Control, Lund Institute of Technology, Barbara Hayes-Roth at the Department of Computer Science, Stanford University, and David Gaba at the Department of Anesthesia, Stanford University, for their help and support in earlier projects. Lars Philipson at the Department of Information Technology is acknowledge for giving the author a chance to start an AI and MFM research group in Lund. Also, I would like to thank Anu Uus for giving several suggestions on how to improve this paper. Finally, great thanks go to Morten Lind, who invented MFM and made all this amazing research possible in the first place.

### References

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

Bengtsson, J. and L.-G. Bäckwall, "Adviser: Computer-Based Decision Support in Intensive Care," Master's thesis, Department of Information Technology, Lund Institute of Technology, Lund, Sweden, 1998.

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.

Chung, D. T. and M. Modarres, "GOTRES: An Expert System for Fault Detection and Analysis," *Reliability Engineering and System Safety*, vol. 24, pp. 113–137, 1989.

Dahlstrand, F., "Alarm Analysis with Fuzzy Logic and Multilevel Flow Models", Proceedings of the 18<sup>th</sup> Annual International Conference of the British Computer Society Special Group on Expert Systems, ES98, Cambridge, England, pp.173-188, 1998.

Dahlstrand, F., "Consequence Analysis Theory for Alarm Analysis", Proceedings of the 2<sup>nd</sup> International Symposium on Engineering of Intelligent Systems, Paisley, Scotland, 2000 a.

Dahlstrand, F., "An Architecture for On-Line Supervision", Proceedings of the IFAC Symposium on Artificial Intelligence in Real-Time Control, Budapest, 2000 b

Dahlstrand, F., "Methods for Alarm Reduction with Multilevel Flow Models of Industrial Processes", Licentiate thesis, Department of Information Technology, Lund University, Lund, 2000 c.

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.

Dubowik, K., "Automated Arrhythmia Analysis—An Expert System for an Intensive-Care Unit," Master's thesis, Department of Information Technology, Lund University, Lund, 1999.

Duncan, K. D. and N. Prætorius, "Flow Displays Representing Complex Plant for Diagnosis and Process Control," Proceedings of the 2<sup>nd</sup> European Meeting on Cognitive Science Approaches to Process Control, Siena, 1989.

Duschek, J., "Syntax Analysis of Modeling Languages and a Knowledge-Based System for Modeling," Master's thesis, Institute of Automatic Control Systems, Technical University of Denmark, Lyngby, 1991.

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.

Jalashgar, A., *Identification of Hidden Failures in Process Control Systems through Function-Oriented System Analysis*, Doctor's thesis, Institute of Automatic Control Systems, Technical University of Denmark, Lyngby, 1997.

Jørgensen, S. S., *Generic MFM Models for Use in Fault Diagnosis of Ship System's Machinery*, Doctor's thesis, Institute of Automatic Control Systems, Technical University of Denmark, Lyngby, 1993.

Kim, I. S. and M. Modarres, "Application of Goal Tree-Success Tree Models as the Knowledge Base of Operator Advisory Systems," *Nuclear Engineering and Design*, vol. 104, pp. 67–81, 1987.

Larsen, M. N., "Modeling Start-Up Tasks Using Functional Models," Final report, Project 4937–92–08–ED ISP DK, CRC JSP Ispra, 1993.

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 9<sup>th</sup> International Symposium on System, Modeling, Control, Zakopane, Poland, 1998.
- 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.
- Lind, M., "Human-Machine Interface for Diagnosis Based on Multilevel Flow Modeling," Proceedings of the 2<sup>nd</sup> 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.
- Modarres, M. and T. Cadman, "A Method of Alarm System Analysis for Process Plants," *Computers and Chemical Engineering*, vol. 10, pp. 557–565, 1986.
- 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.
- Mårtensson, T., "Structuring of Instrumentation and Control Systems Based on MFM," Master's thesis, Department of Information Technology, Lund Institute of Technology, Lund, 1998.
- Öhman, B., "Failure Mode Analysis Using Multilevel Flow Models," Proceedings of the 5<sup>th</sup> European Control Conference, Karlsruhe, Germany, 1998.
- 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.
- Osman, A., *Graphical Control Environment (Grace)*, Doctor's thesis, Institute of Automatic Control Systems, Technical University of Denmark, Lyngby, 1992.
- Price, C. J., D. R. Pugh, N. Snooke, J. E. Hunt, M. S. Wilson, "Combining Functional and Structural Reasoning for Safety Analysis of Electrical Designs," *Knowledge Engineering Review*, vol. 12, no. 3, pp. 271–287, 1997.
- 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.