GENERAL KNOWLEDGE STRUCTURE FOR DIAGNOSIS

Tor Steinar Brendeford, MSc
Institutt for energiteknikk
OECD Halden Reactor Project
Halden, Norway

ABSTRACT

At the OECD Halden Reactor Project work has been going on for several years in the field of automatic fault diagnosis for nuclear power plants. Continuing this work, studies are now carried out to combine different diagnostic systems within the same framework. The goal is to establish a general knowledge structure for diagnosis applied to a NPP process. Such a consistent and generic storage of knowledge will lighten the task of combining different diagnosis techniques. An integration like this is expected to increase the robustness and widen the scope of the diagnosis. Further, verification of system reliability and on-line explanations of hypotheses can be helped. Last but not least there is a potential in reuse of both specific and generic knowledge. The general knowledge framework is also a prerequisite for a successful integration of computerised operator support systems within the process supervision and control complex. Consistency, verification and reuse are keywords also in this respect. Systems that should be considered for integration are; automatic control, computerised operator procedures, alarm - and alarm filtering, signal validation, diagnosis and condition based maintenance.

This paper presents three prototype diagnosis systems developed at the OECD Halden Reactor Project. A software arrangement for process simulation with these three systems attached in parallel is briefly described. The central part of this setup is a 'blackboard' system to be used for representing shared knowledge. Examples of such knowledge representations are included in the paper. The conclusions so far in this line of work are only tentative. The studies of existing methodologies for diagnosis, however, show a potential for several generalisations to be made in knowledge representation and use.

1. INTRODUCTION

The work on plant surveillance systems including diagnosis has a long history at the OECD Halden Reactor Project (HRP). A variety of diagnosis techniques and applications have been implemented since the late eighties. In the past few years the work on diagnosis has been aimed at combining different techniques and methods. There are two major objectives of this work:

- Improved performance and reliability of automatic diagnosis applied to NPP processes.
- More efficient design process for knowledge-based operator support systems, and thereby an increased employment of these systems in the NPP industry.

Core tasks in diagnosis include a search for and presentation of valid hypotheses with fault explanations. Solutions to such tasks have been demonstrated successfully in the previous studies. There is, however, still room for improvements and tuning of the methods. The design process has been addressed, but no final methodology has been chosen. The experience from system developments in the project demonstrates the need for guidelines, or a toolbox, for diagnosis system design. Also different aspects of system integration have been studied but still more work is to be done before a concept can be endorsed.

At the moment the direction pursued is an in-depth investigation of the basis for diagnosis. Diagnosis is bounded by accessible relevant knowledge. Therefore an effort to improve diagnosis could be made by coordinating a search for and a representation of knowledge. In diagnosis of rotating machinery elicitation of operational data is thought to be the major bottle-neck [1]. In other domains the dominant problems of fault diagnosis are related to representation and use of knowledge. Recognising these problems it has been decided to direct the work towards a general representation of knowledge utilised in diagnosis. This raises two questions; what knowledge is used in diagnosis and how to make a general representation.

There are two aspects for which such a knowledge storage is thought to be of use. First it is meant to serve the *diagnosis process* as such, and its application to the plant in question. A consistent and generic storage of knowledge will greatly assist in combining different diagnosis techniques. This will increase the robustness and widen the scope of the diagnosis. Further, verification of system reliability and on-line verification of hypotheses can be facilitated. Last but not least there is a big potential in reuse of both specific and generic knowledge. The second aspect is *integration of computerised operator support systems* within the process supervision and control complex. A general knowledge framework is a prerequisite for a successful combination of the design tasks, like extraction of domain knowledge, and merging the run-time systems. Consistency, verification and reuse are keywords also in this respect.

The next part of this paper gives an overview of three different diagnosis systems. Two different kinds of knowledge represented in these systems are identified. Section 3 presents a simulation set-up including the three diagnosis systems. A central module of this simulation setup is meant to hold some of the knowledge found in the diagnosis systems. The tool and methodology used in making this module is under development in a bilateral project at HRP [2]. Examples of how this tool can be used for knowledge representation are given. The intended outcome of simulations with the set-up is also related. Conclusions and future work constitute the last part of the paper.

2. PROTOTYPE DIAGNOSIS SYSTEMS

Several prototype diagnosis systems have been developed at the OECD Halden Reactor Project. Methods and techniques range from rule-based, model-based, Goal-Tree Success-Tree, to artificial neural networks and fuzzy logic. The developments have followed the historical evolution of methods in the field of artificial intelligence. Often the work has been performed in co-operations with other centres of research. Three diagnosis systems are presented here. Examples of knowledge used in these three systems to diagnose a specific failure are also included. The three systems are studied to find what knowledge is stored in their structures.

2.1 DISKET

DISKET was developed in the mid-eighties, in cooperation with Japan Atomic Research Institute (JAERI) [3], and has been thoroughly tested in the HAlden Man-Machine LABoratory (HAMMLAB) [4]. This is a *rule-based* system containing knowledge of transient behaviour to expand the diagnosis beyond the traditional steady-state process application of rule-based systems.

Fault detection is made by comparing measurements to alarm limits. These alarm limits are given by four qualitative values or by percentages. The hypotheses are organised by physical location in an hierarchical structure where the lower levels pinpoint the origin of failure. In this way isolation and identification are combined. To move through this tree there are rules of stronger and stronger fault explanations. The strength of each explanation is presented explicitly by probability factors. A diagnosis is made whenever the confirmed explanations of a hypothesis have a combined probability factor exceeding a certain limit.

Originally DISKET was coded in FORTRAN but a limited G2-implementation [5] has been made and is used in the current research. In the G2 knowledge base all faults are marked with global identifiers. These are also applied to the two other diagnosis systems presented in this paper. This G2 implementation of DISKET is meant to recognise nine hardware and controller faults in the High Pressure Preheater (HPP) unit described in section 2.4. One of these hardware faults is *heat exchanger 1 (HX1) tube leakage*. Water from the main water flow is leaking into the heat exchanger tank. A DISKET diagnosis of this fault is made from the following sequence of observations:

1 One or more of the heat exchanger tank level measurements of the HPP-unit are registered

with a value above the safety limit.

2 One or more of the heat exchanger tank outlet valves are more than 95% open.
OR
One or more of the heat exchanger tank level measurements of the HPP-unit are registered with a value above the safety limit.

3 The level measurement of HX1 is registered with a value above the safety limit, *and* the tank outlet valve of HX1 is more than 95% open.
OR
The level measurement of HX1 is registered with a value above the safety limit within 60 seconds after the outlet valve of HX1 is more than 95% open.

2.2 EFD/DD

Early Fault Detection (EFD) and Detailed Diagnosis (DD) were developed in consecutive order, as respectively *fault detection* and *fault identification* systems. In combination the two modules perform *model-based* fault diagnosis. The modules, and several applications were made at the OECD Halden Reactor Project during the years 1985-1993 [6] [7].

EFD is based on a partitioning of the target process with a mathematical model of each subprocess. Fault detection is performed by a continuous matching of measurements and predictions relating to each subprocess. Discrepancies between the predictions and the observed behaviour are reduced to a qualitative range of three values. The predictions are made with redundant observations, enabling verification of some sensors. Fault hypotheses are defined in the DD knowledge base. The hypotheses are grouped according to the subsystems modelled in EFD. Fault identification is made in two steps, first the discrepancies found by EFD are matched with symptom patterns stored in DD. The next step use parameter gradient information to select a most probable hypothesis based on the fault candidates identified by the proceeding stage.

The EFD module is programmed in FORTRAN, receiving input from the NOkia Research Simulator (NORS). Corresponding applications of DD have been implemented on a Symbolics LISP workstation and in G2. The G2 version of DD include 35 different fault hypotheses for hardware, controller and sensor failures of the HPP section illustrated in section 2.4. The fault model related to diagnosis of *HX1 tube leakage* in this system include the following observations:

1 The measured outlet flow from the tank of HX1 is not consistent with the flow calculated from heat and mass balances based on other measurements.
2 The measured tank level of HX1 is rising, and the tank outlet valve of HX1 is opening.

2.3 MOAS II

Another *model-based* diagnosis system was developed at the OECD Halden Reactor Project with the MOAS methodology derived at Maryland University by Modarres and co-workers [8]. The diagnosis system development with this method involves a functional analysis, building Goal-Tree Success-Trees (GTST) to identify key surveillance parameters in the process. Knowledge represented in the form of causal graphs is also used in the system design, illustrating plausible causes of observed behaviour. The application created at the OECD Halden Reactor Project was named MOAS II [9].

In MOAS II a small set of measurements are used to detect faults and trigger diagnosis. These measurement values are verified by equations relating redundant sensor information. If any measurements fail to be verified the fault is assumed to be caused by a failing sensor. Isolation and identification of these faults are made by systematically applying mathematical relations interrelating the set of involved measurements. All other faults are identified by first checking weak explanations derived from cause-effect graphs pointing to hypothesis candidates. Secondly strong explanations are used to weed out impossible candidates.

G2 was used for the original implementation, allowing graphical presentation of the diagnosis inference. The present knowledge base contains 48 hypotheses of hardware, sensor and controller failures for the HPP section. However some of the hypotheses are only present in a diagnosis as one of two possible fault explanations. Again we present the inference leading to a diagnosis of *HX1 tube leakage*:

1 One or more of the heat exchanger tank level measurements of the HPP-unit are registered with a value above the safety limit.
2 The main measurements of the HPP-unit verify each other.
3 The tank level of HX1 is above the safety limit, or both the tank levels of HX1 and HX2 are above the safety limit, or all the tank levels of HX1, HX2 and HX3 are above the safety limit.
4 The measurements of the main water flow into HX3 and out of HX1 are not the same, and the calculated mass balances of the HX1 tank are not consistent.

2.4 Demonstration domain

The High Pressure Preheater (HPP) process unit has been chosen to demonstrate the principles and methods of diagnosis. This unit has been used extensively in previous demonstrations and applications developed at the Halden Reactor Project and was selected for the present study because of this. The process is simulated by the NOkia Research Simulator (NORS) installed in HAMMLAB. The NORS process is almost identical to the IVO nuclear power plant (PWR, 500MW), Loviisa, Finland.

The HPPs increase the overall efficiency of the NPP by heating the water before it enters the steam generator in the secondary loop. The unit consists of three heat exchangers coupled in series. The main water flow is supplied at a high pressure from the feed water tanks. Steam is retrieved from bleeding points of the steam turbines, while the hot water comes from a tank. Figure 1 sketch the input-output layout and main components of the HPP section.

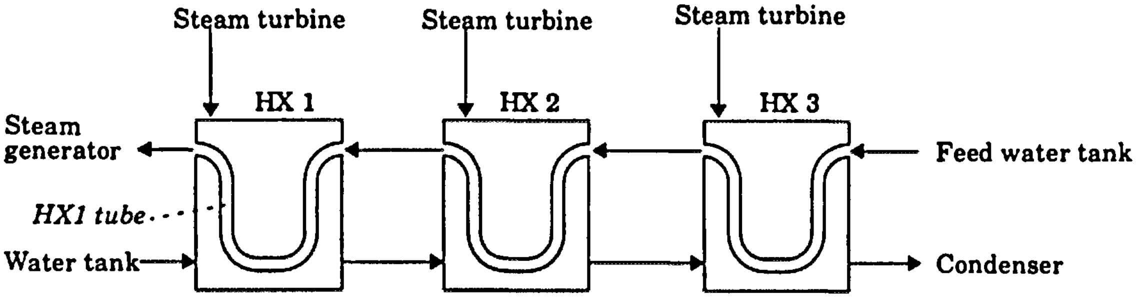

Figure 1. High Pressure Preheater

Some possible fault situations in this heat exchanger network are; leakage of main water into heat exchanger tank, non-functioning control valves, failing sensors and leakages from tubes and tanks to the environment.

2.5 Knowledge used in diagnosis systems

Two types of knowledge can be said to exist in the diagnosis systems described above. Most apparent of the two is the process knowledge like power plant data. A major part of the knowledge used in making a diagnosis is of this kind. This knowledge can be described in terms of the observable parameters, process behaviour, topology and faults related to the application domain of the diagnosis system. The other important type of knowledge in the diagnosis systems is the knowledge of reasoning on how to make a diagnosis. This knowledge is used when making new diagnosis systems, verifying diagnoses and mapping capabilities, strengths and weaknesses of diagnostic methods. This knowledge is used for applying the process knowledge in diagnosis. Examples of diagnostic knowledge are knowledge of backward and forward chaining methods, the detection task, reliability of diagnoses from a diagnosis system, scope of the diagnosis system, and knowledge of methods for identifying key surveillance parameters used in the detection operation.

3. COMPILING KNOWLEDGE FOR DIAGNOSIS

Abilities and distinctions of several diagnosis systems are to be collected and compared to each other. This is done in order to learn more about the knowledge stored in each particular system. A software setup for simulations has been made to facilitate such a knowledge acquisition. The setup include concurrent operation of the three systems DISKET, EFD/DD and MOAS II described in section 2. The diagnoses resulting from the simulations are reported to the central module of the setup. This module is made in a 'blackboard' system referred to as *the BlackBoard*. Creating this application of the BlackBoard is the first step in generating a general knowledge framework for diagnosis. The BlackBoard tool is under construction in a separate project at the HRP [2]. In addition to the connections to the three diagnosis systems the BlackBoard is also linked to a program performing an analysis of the reported diagnoses. The software programs and their interfaces are illustrated in Figure 2.

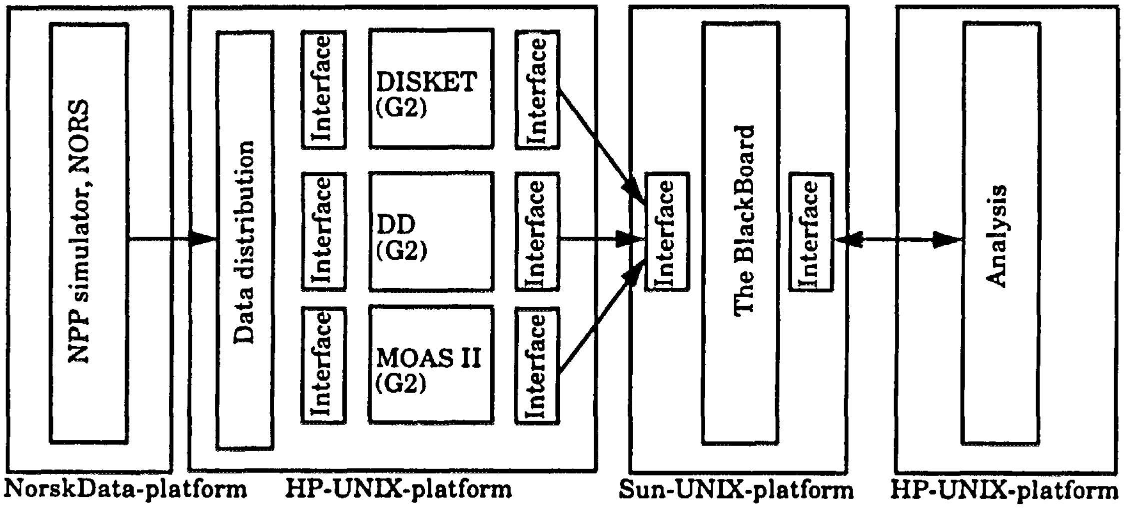

Figure 2. Setup for parallel diagnoses

First we will give a description and examples of how to represent knowledge on the BlackBoard. Then features of the simulation setup regarding compilation of knowledge is related.

3.1 The BlackBoard

The purpose of the BlackBoard is to coordinate and communicate knowledge coming from diverse knowledge sources. A knowledge source could be almost any form of code describing a system or procedure. Typically the knowledge sources work together to solve a problem. This requires the BlackBoard to represent knowledge in an acceptable format, and to transfer knowledge to and from a multitude of agents. The analogy to a blackboard used at school is good. Envision a group of people gathered around the blackboard to solve a problem. They are allowed only to communicate by writing or erasing on the blackboard.

Representation of knowledge on the BlackBoard is facilitated by objects. Object classes, object instances, object attributes and object relations are to be made for the domain in question. Lists, originating objects and special relations are used to keep track of the elements on the BlackBoard. Communication with external entities is enabled by functions working on the structure of objects. Another important feature of the BlackBoard is the ability to restrict access to knowledge. Depending on the access level of a knowledge source certain parts of the knowledge domain are hidden or revealed.

How to represent knowledge on the BlackBoard is to a large extent based on the domain itself and what the knowledge is to be used for. There are indeed very few rules or limitations, allowing the BlackBoard to be applied to a broad range of system domains. This flexibility can on the other hand lead to lack of conformity. To secure some consistency the BlackBoard is delivered with a set of defined relations. One of these is the 'is-a' relation, made to connect object classes and instances. However, the BlackBoard does not define semantics associated with relations [10]. As a consequence it is difficult to create good ontologies for the domains on the BlackBoard. Here ontology means *specification of a conceptualization*, a definition applied in the context of knowledge sharing [11].

What knowledge to store on the BlackBoard is partly a question of what knowledge should be shared between multiple agents and what is better left to be presented in the knowledge sources. Other issues affecting what knowledge to represent on the BlackBoard concern the original knowledge format and the software interfaces of the BlackBoard.

Technically the BlackBoard exists as a C++ program. Libraries contain definitions and functions to create domains for the BlackBoard and connections to other knowledge sources. Parallel Virtual Machine (PVM) software is used for creating the server-client architecture [12]. A graphical interface based on Gain Momentum is being developed [13].

3.2 Process knowledge on the BlackBoard

Concepts and particular process knowledge of a NPP are needed in diagnosis systems for the plant. Much of the data are common to more systems and should therefore be presented on the BlackBoard. This would also allow diagnosis systems of different methods to share the knowledge of a specific NPP. There are already a multitude of ontologies and representational techniques made for process knowledge of the NPP domain. However, few of these offer simple application, verification and reuse in computerised operator support systems.

An example of how topological knowledge can be stored on the BlackBoard is given in Figure 3. This knowledge is of course again connected to other kinds of knowledge in the diagnosis systems. In Figure 3 there is both system independent (generic) knowledge, presented in the form of classes, and system specific knowledge, as objects, from the NPP domain. System dependent topological knowledge, like the class instances and relations in Figure 3, is used in isolating the location of faults.

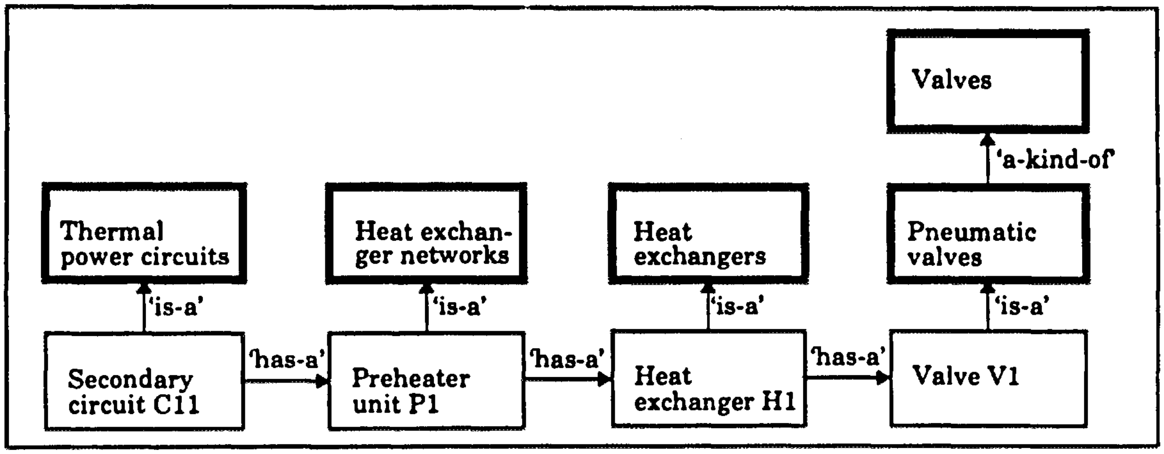

Figure 3. Example of generic and specific topological knowledge

A second example on how to represent process knowledge on the BlackBoard displays elements of process behaviour. There are many ways to represent such knowledge. Figure 4 illustrates an approach with *function* objects. Other commonly used representations present behaviour as *events* or *states*. In portraying knowledge of behaviour it is pertinent to show the casual relations between the functions, events or states. These relations illustrate the connections creating the global operation.

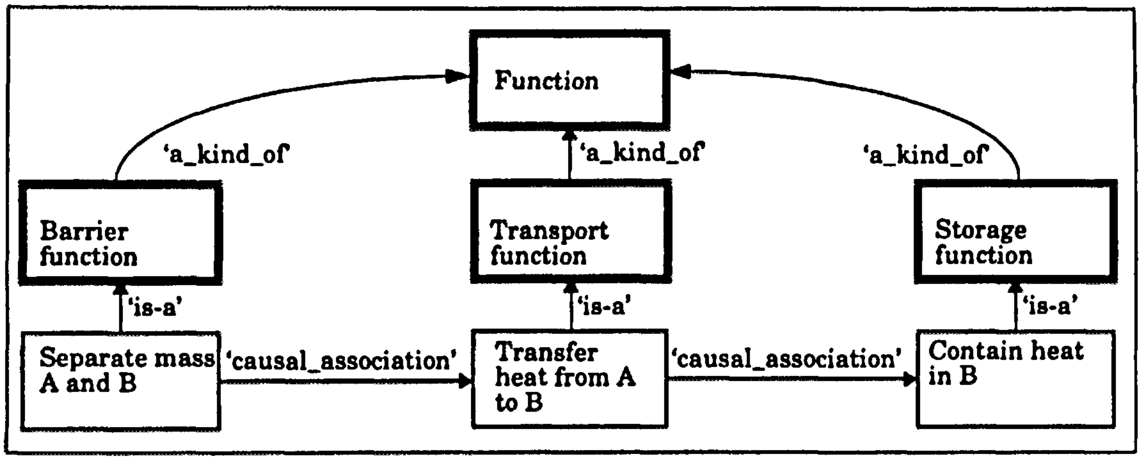

Figure 4. Behavioural knowledge by functions

3.3 Knowledge of diagnostic reasoning on the BlackBoard

In the AI field diagnosis methods are divided into two broad classes the *model-based* and *association-based* methods. Such a classification is for example made in the CommonKADS Library [14], where the two methods are called *consistency-based* and *association-based*. Inspired by the KADS models Figure 5 visualize some ideas for presenting knowledge of these methods on the BlackBoard. However, any detailed frameworks encompassing all the aspects of what we call diagnostic knowledge have not yet been found, and it is therefore hard to make examples of representations on the BlackBoard. The non-physical nature of diagnostic knowledge is also making it less intuitive to think of appropriate classes and objects to be placed on the BlackBoard to illustrate this knowledge.

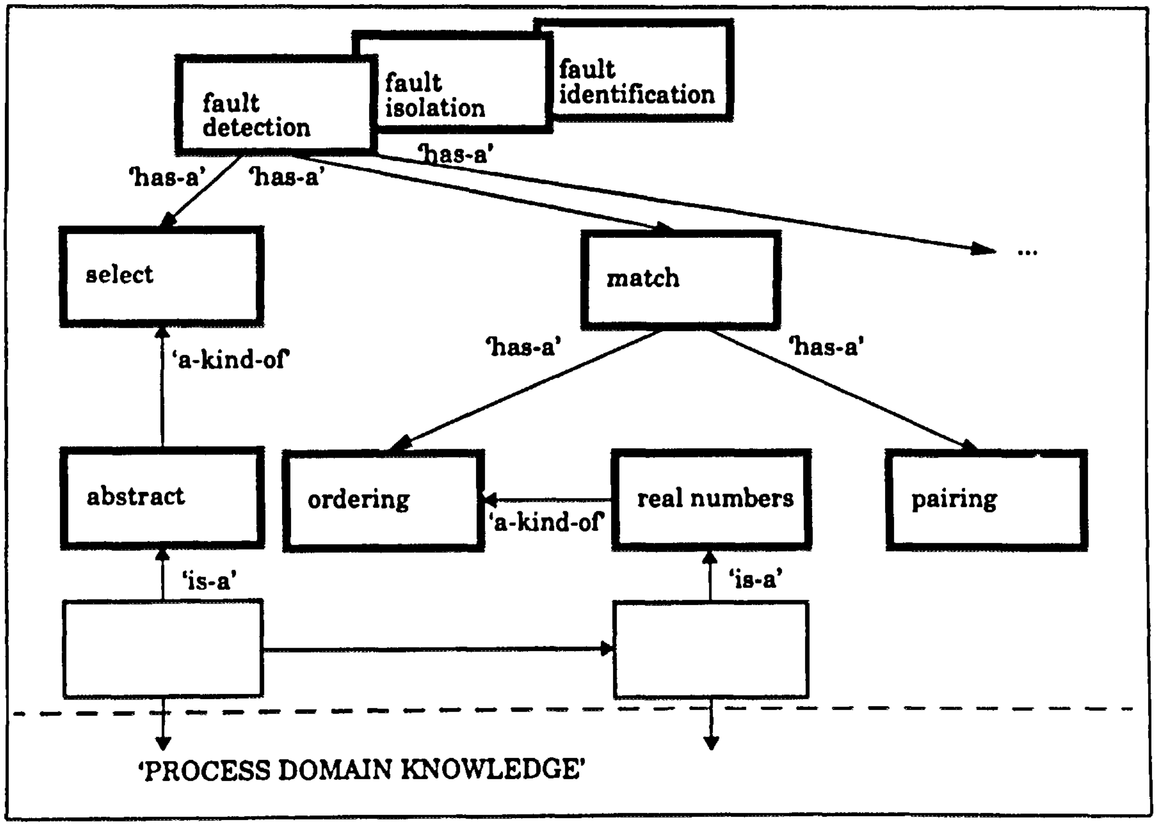

Figure 5. Concepts used in association- and model-based diagnosis

3.4 Simulation-based learning

A simple architecture for knowledge used in diagnosis has been implemented on the BlackBoard, see Figure 6. The main intention of this structure is to allow compilation of knowledge from diagnosis systems, but it is also thought to be the starting point for a framework of knowledge used in diagnosis. Both process knowledge, like the hypotheses related to the process in question, and diagnostic knowledge represented by for instance the 'Hypotheses' class are shown in Figure 6.

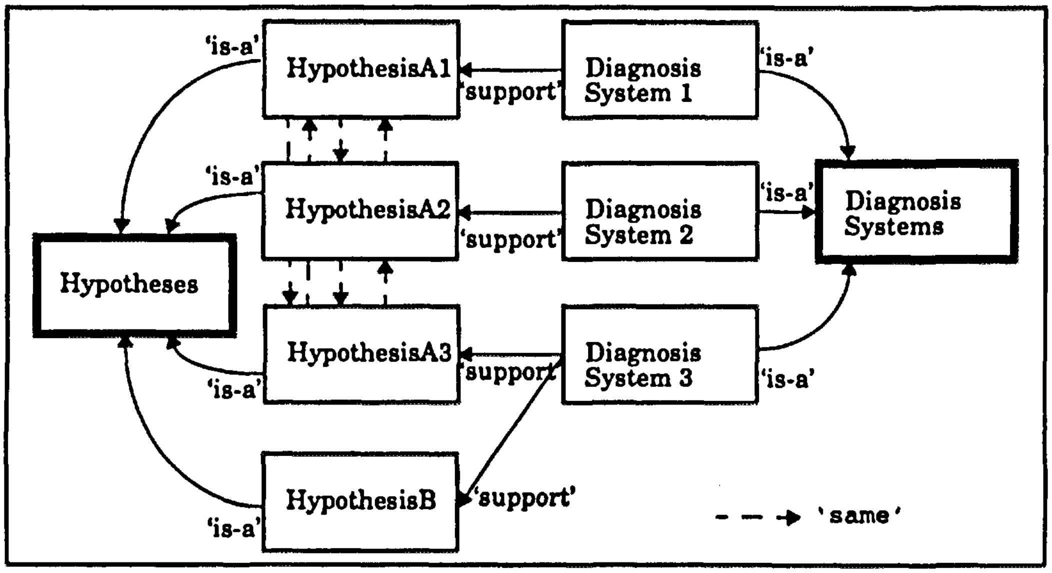

Figure 6. Basic BlackBoard domain model for diagnosis

The intended compilation of knowledge require an analysis of scenario results collected on the BlackBoard. The actual simulation scenarios have not yet been made or run with this setup, but an analysis module is developed in order to assist in evaluating the diagnoses reported to the BlackBoard. This analysis module is connected to the BlackBoard as a knowledge source with two-ways communication as shown in Figure 2. Only a simple form of reasoning is currently performed by this knowledge source. The diagnoses reported to the BlackBoard are compared in order to find whether all the diagnosis systems report the same fault and at what point in time the faults are diagnosed. Extensions are necessary both in the BlackBoard domain (Figure 6) and this analysis module to investigate other features of the diagnosis systems in the simulation setup.

The knowledge attained from the simulations is of course limited to the knowledge stored in the connected diagnosis systems. Hopefully it will be possible to acquire both process and diagnostic knowledge through the simulations. The current setup is expected to tell us what hypotheses are the same in the different diagnosis systems, what hypotheses overlap and maybe which ones are contradictory. Simple knowledge of the scope and reliability of the particular systems can perhaps also be retrieved by investigating the results of many simulation scenarios.

4. CONCLUSIONS

Three prototype diagnosis systems are described by their applications of knowledge and examples of knowledge related to a particular fault. These examples indicate a similarity in the actual knowledge used by the three different systems. A better comparison and possible integration of the systems or their methods require a more thorough analysis of the systems. Such an analysis is thought to be facilitated by simulation. A simulation setup has been made and is presented with the three diagnosis systems attached to it.

The central module of the simulation setup is made with a 'blackboard' system named *the BlackBoard*. The description of the BlackBoard and the examples of representations of knowledge for diagnosis illustrate how the BlackBoard can be utilised in creating a general knowledge structure for diagnosis. Recalling our intentions this structure is expected to assist in improving both the design process of diagnosis systems and the diagnosis process itself by supporting reuse, integration and verification.

Future work include running simulations with the setup of diagnosis systems. This is hoped to give both new insight in the particular diagnosis systems and diagnosis system knowledge in general. New prototype diagnosis systems can be connected to the setup for testing and investigation. The lessons learned from the simulations will also assist in expanding the knowledge structure on the BlackBoard. Extensions to this structure are to be made both for process and diagnostic knowledge.

REFERENCES

[1] Egusquiza, E. and White, M.F., 1992. *Analysis and Diagnosis of Pump Problems.* Technical Diagnosis, Budapest IMEKO 1992, 8th International Symposium, September, 1992

[2] Hamon, Leroy, Spaggiari, Bremdal, Grini and Nilsen, 1996. *Tool for Capitalizing and Utilising Knowledge.* To be presented at the OECD Halden Reactor Project EHPG Meeting, Loen, Norway, 1996

[3] Yokobayashi, M., Grini, R-E., and Berg, Ø., 1987. *Documentation of the Diagnosis System DISKET.* OECD Halden Work Report (HWR) 188, Institutt for energiteknikk, Halden, Norway, February, 1987

[4] Endestad, T., Holmström, C.B.O., and Volden, F.S., 1992. *The Second Experimental Evaluation of DISKET: The Diagnosis System Using Knowledge Engineering Technique.* OECD Halden Work Report (HWR) 307, Institutt for energiteknikk, Halden, Norway, August, 1992

[5] G2 Reference Manual, Version 4.0, 1995. Gensym Corporation, Cambridge, Massachusetts, USA, 1995

[6] Berg, Ø., Grini, R-E., Suzudo, T., and Westgaard, T., 1988. *Detailed Diagnosis Coupled to Model-based Fault Detection in Process Plant Operation.* OECD Halden Work Report (HWR) 220, Institutt for energiteknikk, Halden, Norway, May, 1988

[7] Bye, A., and Ness, E., 1993. *Early fault detection and on-line diagnosis in real-time environments.* Modeling, Identification and Control, Vol.14, no. 1, p. 27-41, 1993

[8] Kim, I.S., Modarres, M., and Hunt, R.N.M., 1990. *A Model-Based Approach to On-Line Process Disturbance Management: The Application.* Reliability Engineering and System Safety (29), p. 185-239, 1990

[9] Kim, I.S., Grini, R-E., and Nilsen, S., 1994. *A New Surveillance and Diagnosis System for NORS Based on the MOAS II Methodology.* OECD Halden Work Report (HWR) 386, Institutt for energiteknikk, Halden, Norway, October, 1994

[10] Nilsen, S., 1995. *IDS BlackBoard Application.* OECD Halden Reactor Project, Control Room Division, Internal document, MMSR-2055, Halden, Norway, August 7th, 1995

[11] Gruber, T.R., 1993. *A translation approach to portable ontologies.* Knowledge Acquisition, 5(2):199-220, 1993

[12] Geist, A., Beguelin, A., Dongarra, J., Jiang, W., Manchek, R., and Sunderam, V., 1994. *PVM: Parallel Virtual Machine - A Users' Guide and Tutorial for Networked Parallel Computing.* The MIT Press, Cambridge, Massachusetts, 1994

[13] Gain Momentum User's Guide, 1994. Release 3.0-41. Sybase Inc., Paolo Alto, California. USA, 1994

[14] Breuker, J. and Van de Velde, W.(eds.), 1994. *The common KADS library for expertise modeling: Reusable Problem Solving Components.* IOS Press, 1994