AND/OR/NOT CAUSAL GRAPHS — A MODEL FOR DIAGNOSTIC REASONING[†]

Antoni LIGĘZA*, Pilar FUSTER PARRA**

The paper addresses the issues of diagnostic reasoning based on abductive analysis of causal structures. It is based on ideas emerging from an engineering approach to diagnosis of complex systems. A diagnostic process is considered as a multistage backward search procedure supported by sequential testing and ordering strategies. A basic, core, and uniform model for representing causal behaviour of diagnosed systems is proposed; it has the form of an AND/OR/NOT causal graph allowing for the specification of causality types reflecting the basic logical operations. A multistage approach to diagnostic reasoning is outlined. The discussion covers initial failure detection (with the use of an expected-behaviour approach rather than the complete model of the system), diagnostic reasoning based on search of the graph, and a final validation stage of generated possible diagnoses. Finally, possibilities of further extensions and related work are pointed out.

1. Introduction

Automated diagnosis constitutes an important area of both applied Artificial Intelligence (AI) and modern control theory. The main aim is to support the human operator by means of a symbolic representation of knowledge and reasoning about faulty behaviour of technical systems. The goal of diagnostic reasoning is to detect (detection of abnormal behaviour) and to determine (isolate) faults being the initial causes of abnormal behaviour.

There are a number of approaches to the formal statement and implementation of diagnostic procedures. The most popular ones include approaches based on analytical models for fault detection (Isermann, 1993; 1994; Frank and Köppen-Seliger, 1995), methods based on pattern recognition and neural networks (Korbicz *et al.*, 1994), AI and expert systems (shallow-knowledge-based) methods, and (deep) model-based AI approaches (Davis and Hamscher, 1992; Struss, 1992); for some examples and classifications see also (Korbicz, 1995; Korbicz and Cempel, 1993; Ligęza and Fuster

[†] The research was supported by the Polish State Committee for Scientific Research under grant No. 8 T11A 013 08.

* Institute of Automatics AGH, al. Mickiewicza 30, 30–059 Kraków, Poland, e-mail: ali@ia.agh.edu.pl.

** Supported by the University of Balearic Islands, Cra. de Valldemossa, km. 7.5, E–07071 Palma, Spain, e-mail: pilar@ipc4.uib.es.

Parra, 1994; Ligęza *et al.*, 1996; Saucier *et al.*, 1989; Torasso and Console, 1989; Tzafestas, 1989).

A diagnosis is usually regarded as fault detection and localization (Frank and Köppen-Seliger, 1995; Isermann, 1994; Korbicz *et al.*, 1994). However, some approaches consist in detection and recognition of fault type only. This is typical for analytical model-based procedures, and neural network and pattern-recognition-based methods; the main idea of such methods consists in determining and mapping the current state into its qualitative classification (e.g. *normal* or *faulty*). Such methods are usually based on an implicit assumption of numerical character of data, measurability of signals, functional dependencies between signals and accessibility of complete information; in consequence, they can be applied for a limited class of systems and, in fact, they perform only the detection and classification stage of a diagnostic process.

A complete diagnostic procedure must employ means for fault localization, and thus some form of inference should be performed. AI offers several solutions, but present-day supervision, diagnostic and control systems are implemented mainly as rule-based expert systems using shallow diagnostic knowledge of the experts (Isermann, 1994; Tzafestas, 1989). The implementation and debugging of such systems is a time-consuming and very tedious task. The knowledge acquisition problem is still a bottleneck for many potential applications. Furthermore, the performance of such systems is limited to the class of problems described by the rules acquired from an expert.

In order to overcome the above-mentioned difficulties of the first-generation diagnostic expert systems, another approach is put forward by AI (Davis and Hamscher, 1992; Saucier *et al.*, 1989; Struss, 1992). The very basic idea of modern approaches consists in representing the internal structure and behaviour of systems rather than the shallow diagnostic knowledge. Determination of faults is assumed to be done with an appropriate reasoning mechanism applied to this deep knowledge and causal dependencies specified. This kind of approach is also referred to as *the model-based diagnosis* or *diagnosis from first principles* (Reiter, 1987), and it seems to constitute an emerging AI technology for dealing with failures of complex systems. Some exemplary approaches of this kind are presented in (Console and Torasso, 1988; 1992; Console *et al.*, 1989; Davis, 1984; 1993; DeKleer and Williams, 1987; Genesereth, 1984; Reiter, 1987; Torasso and Console, 1989).

There is no unique and well-defined methodology for knowledge-based diagnosis. The basic approaches under consideration split into two main categories, i.e.:

- consistency-based approaches,
- approaches based on abductive reasoning.

The basic idea of approaches based on consistency verification consists in an analysis of a complete model of system behaviour with respect to keeping its logical consistency (Reiter, 1987). The model must be 'parameterized' with respect to possible faults of elements. If a failure occurs, the observed abnormal behaviour is inconsistent with the knowledge about the system and the assumption that all its components are correct. Then, to regain consistency, some of the components must be assumed to be faulty; such an assumption leads to generation of possible diagnoses. This explains the notion of *consistency-based* approaches.

On the other hand, abductive reasoning consists in finding an explanation for given observations with the use of inference rules based on causal dependencies. In the case of abduction, the rules are interpreted backwards. Abduction constitutes an inference mode opposite (inverse) to deduction. In contrast to deduction, abduction is not a valid logical inference rule. However, it seems to be the closest to human diagnostic reasoning. Supported with the possibility of testing and verification and guided by expert knowledge, it may constitute a powerful tool for diagnostic reasoning.

In this paper, a more complex understanding of diagnostic processes is accepted. It is oriented towards integrating monitoring, failure detection, fault diagnosis and diagnoses verification. A *diagnostic process* is considered here as a complex, multi-stage and multiple-approach based activity aimed at failure detection, localization of faulty components and repair. Thus a diagnostic process is assumed to comprise at least the following three main stages:

- detection and classification of a failure,
- search for faulty components responsible for the abnormal behaviour observed; during the search further complex auxiliary activities can take place, such as hypothesis formation and focusing attention, heuristic search ordering, testing, hypotheses rejection and search restriction, etc.,
- final verification of potential diagnoses found and repair.

The main goal of the paper is to propose a model of diagnostic reasoning based on the assumption that *search* and *abductive reasoning* are the key tools for diagnosing more complex systems. In the case of lack of domain diagnostic experience, but when some knowledge about the system elements, their behaviour, and mutual interactions between them is accessible, a causal model seems to constitute the most appropriate tool for diagnosis. In contrast to logic-based approaches, it is not assumed that a complete model of the diagnosed system is accessible.

As the main contribution, a proposal for a general, core model to represent the causal knowledge used in diagnostic inference in the form of an appropriate graph (called an AND/OR/NOT causal graph) is put forward. Such a graph can serve as a basic tool for diagnostic inference, as it defines the *search space* for fault localization. At the same time, it provides a possibility of implementing several auxiliary mechanisms for enhancing the efficiency of diagnostic reasoning. Moreover, it seems to be close to the engineering way of diagnostic analysis, and thus provides an intuitive and easily understandable tool for domain experts.

Three main ideas are prevailing in the present paper. Firstly, it is pointed out that the *failure detection* can be and usually is performed without a great deal of knowledge about the system structure and components. In practical cases, this can be done with reference to the knowledge about *the expected behaviour* or expected output to be observed. This detection of abnormal behaviour constitutes the first step of a diagnostic process.

Secondly, it is claimed that diagnostic reasoning is a process based on *search* rather then any other kind of inference (e.g. simple input-output mapping of classification type). The search, however, is supported by a number of tools and techniques making it a sophisticated intellectual activity (provided it is not a routine diagnosis performed by a predefined technical diagnostic procedure) rather than a pure systematic or ordered graph-search procedure.

Thirdly, it is assumed that any diagnostic process is, in fact, a *multistage* and *multiple-approach* based process. The most important stages are *failure detection* (see above), *search for possible explanations*, and finally *validation of obtained diagnoses*. During the second and third stage various auxiliary tools for *testing*, *ordering* and *pruning* of working hypotheses can be applied.

To summarize, the main objective is to put forward elements of a basic, uniform, generic approach to a *multistage* diagnosis of complex technological systems based on deep (but usually incomplete) knowledge about the causal structure and behaviour. Moreover, several extensions and enhancements providing a possibility to improve the efficiency of realistic diagnostic procedures are pointed out. The paper is based on the authors' research presented more completely in (Fuster Pařa and Ligęza, 1996; Ligęza and Fuster Parra, 1994; Ligęza *et al.*, 1996). Some further ideas are also presented in (Fuster Parra and Ligęza, 1995a; 1995b; Ligęza and Fuster Parra, 1995a) and a recent complete work (Fuster Parra, 1996).

2. An Approach to Failure Detection Based on Expected Behaviour

The basic idea is that the diagnosis of a system is usually performed on-line in connection with process monitoring. The diagnostic procedure itself is regarded as a multistage search process of nonmonotonic reasoning. The following surrounding activities are assumed to constitute the complete procedure:

- *monitoring*, including signal-to-symbol transformation in order to arrive at a qualitative-logical description of the current state of the system,
- *fault detection*: determining the current qualitative situation and detecting incorrect behaviour,
- *fault diagnosis*: determining potential faulty components, control actions and operational conditions responsible for observed failures with the use of some causal dependency structure,
- *efficiency improvement*: application of hypotheses formation, tests, ordering of the search and constraints so as to improve the efficiency of the search,
- *diagnoses verification*: elimination of some potential diagnoses by tests and simulation of behaviour of certain components, and comparison of the results with observation.

The main stage of the diagnostic process consists in a search for potential diagnoses explaining the observed abnormal behaviour. Let $D = \{d_1, d_2, \ldots, d_d\}$ denote a set of elementary faults, i.e. initial reasons for possible failures, and let $M = \{m_1, m_2, \ldots, m_m\}$ be the set of *manifestations*, i.e. the outermost symptoms of a failure. It is assumed that there is some causal dependency mapping the set 2^D into the set 2^M; however, the mapping is not given explicitly, not all the information concerning the mapping is provided and the mapping is not a one-to-one function. Thus one cannot expect a simple reconstruction of the inverse mapping which would constitute a solution to diagnostic problem.

In order to provide some example, one can think about a diagnosis of serially connected bulbs (as used for illuminating the Christmas Tree). The failure detection and classification stage consists in observing that the system does not produce light, even when switched on. The core of a diagnostic process consists in a search for bulbs and connections which do not work properly. Verification and repair consist in exchanging the faulty ones and finally observing the correct behaviour, i.e. the light on.

In the case of complex, realistic systems it is likely that a complete logical model of the system will not be achievable any more. Further, majority of human supervisors of dynamic processes seem not to concentrate on the model of the process, but rather on the *expected output* taken alone. This suggests the idea that it may be reasonable to describe in some way what the process is expected to do (i.e. what the expected output behaviour should be) and, on the other hand, what should not happen.

Let us consider a watch as an example. Even if one does not know the principles of working of this complicated device, one is able to say if it works normally or not. Typically, four qualitative situations referring to the output can be distinguished, i.e.:

- the watch is O.K.,
- the watch is not working,
- the watch is fast,
- the watch is slow.

Thus, the first step of a diagnostic procedure can be performed having only some idea about *the expected behaviour* of the system and comparing it with most superficial observations. The same applies to many other, even very complex systems, like TV, a car, a washing machine, an airplane, etc. or even a human organism.

The general idea for failure detection is based on the notion of *expected behaviour* (Ligęza and Fuster Parra, 1994; 1995a). It is assumed that the user is able to recognize various types of abnormal behaviour and classify the observed one into a specific category. This can be done with pattern recognition methods, neural network approaches, or logical reasoning. The notion of expected behaviour includes both *expected normal behaviour* and various sorts of *expected abnormal behaviour*.[1]

[1] In several approaches the notion of *expected behaviour* is used as a reference point for detection of abnormal behaviour, but *expected* is used only with the underlying meaning of a *normal* one; on the contrary, the notion *expected* here means mostly *expected misbehaviour*, i.e. knowledge useful for recognizing the outermost types of failures.

The expected behaviour of the system is assumed to be specified with the use of the manifestations $\boldsymbol{M}$. The current qualitative situation is represented by a selection of two subsets of $\boldsymbol{M}$, namely those detected to be true M^+ and those observed to be false M^-, $M^+ \subseteq \boldsymbol{M}$, $M^- \subseteq \boldsymbol{M}$. The methods of detecting the current values of symptoms from $\boldsymbol{M}$ may consist in observation, monitoring and comparison with predefined reference values, performing diagnostic tests (Kościelny, 1995a; 1995b; Kościelny and Pieniążek, 1994), or logical reasoning (Fuster Parra, 1996; Ligęza, 1995; Ligęza and Fuster Parra, 1995a).

The final detection of a failure is performed by comparing of the current sets of detected manifestations M^+ and M^- with predefined, user-specified qualitative descriptions of the expected abnormal behaviour. In the simplest case, let Q_i^+, Q_i^- denote a pattern of manifestations specifying the i-th possible failure, $i \in \{1, 2, \ldots, q\}$, where q is the number of potential failures. In order to check that the current state of manifestations satisfies the specification of the i-th failure, we have to test if

$$Q_i^+ \subseteq M^+ \quad \text{and} \quad Q_i^- \subseteq M^-$$

If the above conditions are satisfied, the i-th failure takes place, and the diagnostic procedure should be run. Some more detailed description concerning a language for describing qualitative situations and formulae-matching mechanism for detecting failures is presented in (Fuster Parra, 1996; Ligęza, 1995).

3. Representation of Causal Behaviour

3.1. Symptoms

In technical terms a *symptom* is usually used to denote a characteristic of a system occurring at a time instant. The key issue for explaining faulty behaviour in technical systems is the knowledge of *causal* relationships among *symptoms* occurring in the system.

The current status of a symptom can be one of the following three possibilities: *true* (known to occur), *false* (known not to occur), or *unknown*. In this work symptoms are considered rather as a useful linguistic category than as a precisely-defined logical term. However, from a logical point of view, one can often regard them as described by logical formulae.

Several categories of symptoms may be considered. First, there are *external manifestations*, usually denoting *failure symptoms* and indicating abnormal or faulty behaviour of the system under consideration. They constitute the principal observable output of the system under diagnosis. The set of manifestations to be considered will be denoted by $\boldsymbol{M} = \{m_1, m_2, \ldots, m_m\}$.

Second, a set of *initial cause* symptoms is distinguished. An *initial cause* is a symptom without any visible reason or for which one does not search for any further cause or explanation. Initial causes can be divided into several subcategories, e.g. component faults, control actions, external conditions, etc. All of them are considered as *elementary diagnoses*. The set of elementary diagnoses is denoted by $\boldsymbol{D} = \{d_1, d_2, \ldots, d_d\}$.

The third category of symptoms consists of *intermediate* (other) symptoms. These are just any symptoms, observable directly or not, which are neither manifestations ($\boldsymbol{M}$) nor elementary diagnoses ($\boldsymbol{D}$). The set of such symptoms will be denoted by $\boldsymbol{V} = \{v_1, v_2, \ldots, v_v\}$.

The set of all symptoms will be denoted by $\boldsymbol{N}$, where $\boldsymbol{N} = \boldsymbol{D} \cup \boldsymbol{V} \cup \boldsymbol{M}$. Finally, for clarity of the discussion it is assumed that $\boldsymbol{D} \cap \boldsymbol{V} = \emptyset$, $\boldsymbol{D} \cap \boldsymbol{M} = \emptyset$, and $\boldsymbol{V} \cap \boldsymbol{M} = \emptyset$.

Taking into account practical diagnostic problems, any element of $\boldsymbol{N}$ can be regarded as a *symptom*, *event*, *binary variable*, or *propositional formula*. Thus several ways of notation are possible: a symptom $n \in \boldsymbol{N}$ being observed can be denoted by n being *true*, $n = 1$ or simply n, while its absence can be denoted by n being *false*, $n = 0$, or $\bar{n}$.

3.2. Causal Relations

For the sake of graphical representation, it is assumed that causal relations among symptoms are specified with the use of graph representation. Any node of the graph represents a specific symptom $n \in \boldsymbol{N}$. Whenever there is a causal relation between nodes n and n', there is a directed arc pointing from n to n'. The meaning of the arc is that n *causes* n'. When using logical notation, the semantics of a single connection between two nodes is defined by $n \models n'$, i.e. if n becomes *true*, then also n' necessarily becomes *true*. A more detailed definition and some extensions are considered in (Fuster Parra, 1996; Ligęza and Fuster Parra, 1994; Ligęza *et al.*, 1996).

Now, let $\boldsymbol{E}^2 \subseteq \boldsymbol{N}$ denote all such simple dependencies, i.e. $\boldsymbol{E}^2$ is a binary relation representing causal dependencies among pairs of symptoms. In case symptoms $n_1, n_2, \ldots, n_i$ cause n only when occurring simultaneously, one says that the conjunction of $n_1, n_2, \ldots, n_i$ causes n. Let $\boldsymbol{E}^{i+1}$ denote the set (i.e. an $(i+1)$-ary relation) of all such dependencies, where $i \geq 2$. To simplify the notation, let us write $\boldsymbol{E}^* = \boldsymbol{E}^2 \cup \boldsymbol{E}^3 \cup \ldots \cup \boldsymbol{E}^{l+1}$, where l is the maximal number of symptoms (in the system considered) causing simultaneously some symptom to occur. Further, let $\boldsymbol{E}^-$ denote the set of binary negative influences, i.e. $(n, n') \in \boldsymbol{E}^-$ iff the lack of n (i.e. its negative truth value) causes n' to occur.

3.3. AND/OR/NOT Causal Graphs

Let $\boldsymbol{N}$ denote a set of symptoms, $\boldsymbol{N} = \boldsymbol{D} \cup \boldsymbol{V} \cup \boldsymbol{M}$, where $\boldsymbol{M}$ is a set of manifestations (usually, fault symptoms), $\boldsymbol{D}$ is a set of elementary diagnoses, and $\boldsymbol{V}$ is a set of intermediate symptoms to be considered. Further, let $\boldsymbol{E}^*$ denote a set of relations defining causal dependencies, and $\boldsymbol{E}^-$ a set of relations defining binary negative dependencies. A definition of the causal graph defining relationships between the symptoms is presented below.

Definition 1. An AND/OR/NOT causal graph $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{E}^*, \boldsymbol{E}^-)$ is the graph having nodes specified with $\boldsymbol{N}$ and arcs defined by $\boldsymbol{E}^*$ and $\boldsymbol{E}^-$ and satisfying the following conditions:

- there is no arc pointing from the nodes specified with $\boldsymbol{M}$, i.e. the manifestations are final nodes,
- there is no arc pointing to the nodes of $\boldsymbol{D}$, i.e. the elementary diagnoses are initial nodes,
- there are no loops in the graph.

Further, to make the graph connected, it may be assumed that every node has at least one arc pointing to or starting from this node, etc. Further considerations and extensions are discussed in (Fuster Parra, 1996; Ligęza *et al.*, 1996).

The interpretation of the above definition is straightforward. There is a directed arc from node n to node n' whenever n causes n'. It is also possible that there are independent arcs starting from several nodes and pointing to n'. In this case, the occurrence of any of the symptoms represented by these nodes is satisfactory for n' becoming true. In such a case, node n' is referred to as an OR-node. If symptoms $n_1, n_2, \ldots, n_i$ cause n only when they occur together, then all the arcs from $n_1, n_2, \ldots, n_i$ to n are joined by a special "horizontal" arc and form an AND connection. Node n is then referred to as an AND-node. Further, there is an arc labelled NOT from n to n' whenever the absence of the symptom defined by n (its negation) causes n' to occur.

An example of an AND/OR/NOT causal graph is shown in Fig. 1.

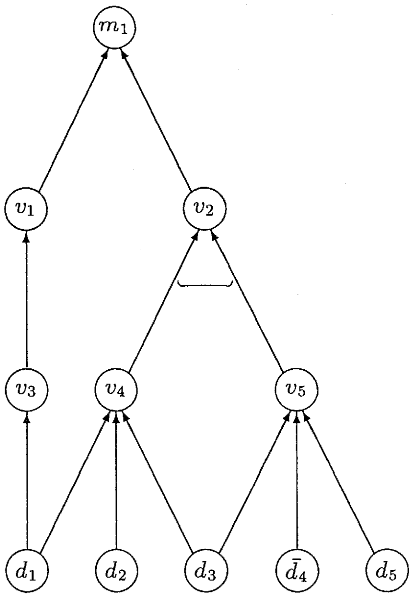

Fig. 1. An example of an AND/OR/NOT graph.

The AND/OR/NOT causal graph defined above is similar in structure to classical AND/OR graphs used in problem solving (Nilsson, 1971). The main difference consists in the direction and interpretation of arcs (here different extensions of the basic causal interpretation are possible; see e.g. the MAY links in (Console and Torasso, 1988; 1992; Console *et al.*, 1989; Torasso and Console, 1989); see also (Ligęza and Fuster Parra, 1994) and nodes—an initial node means here only a *possibility* of solution. The application of the graph is also different. A visible extension consists in admitting the NOT links. Further, a "solution graph" in classical problem solving constitutes only a "possible justification" (to be further validated) for the observed failure. The AND/OR/NOT causal graphs constitute also a refinement of fault trees (Barlow and Lambert, 1975) and causal graphs used in the model-based diagnosis (Chang *et al.*, 1991; Lunze and Schiller, 1992) oriented towards a direct search for diagnoses.

4. A Diagnostic Problem Statement

A diagnostic problem exists if at least one fault is observed. The faults to be diagnosed are assumed to be specified with some manifestations. Let $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{E}^*, \boldsymbol{E}^-)$ be an AND/OR/NOT causal graph. The definition of a diagnostic problem takes also into account possible observations providing further information to the diagnostic system.

Definition 2. A diagnostic problem is a quintuple $(\boldsymbol{G}, M^+, M^-, N^+, N^-)$, where M^+ and M^- are the failure manifestations to be diagnosed, which are respectively true and false, and where N^+ and N^- are the auxiliary observations specifying which further symptoms are true and false. We have $M^+ \subseteq \boldsymbol{M}$, $M^- \subseteq \boldsymbol{M}$, and $N^+ \subseteq \boldsymbol{N}$, $N^- \subseteq \boldsymbol{N}$.

The sets of manifestations of *true* and *false* provide the fault definition and must be explained by all the diagnoses found, while the auxiliary observations provide information which can be used for guiding, ordering and refining the search. Further, any diagnosis must be consistent with the observations.

4.1. Search for Diagnoses and State of the Search

In order to solve a diagnostic problem, we have to search for a set of possible initial cause symptoms explaining (justifying) the failures specified by the current manifestation values. Basically, the search for possible diagnoses (not necessarily the minimal ones) can be any systematic graph search procedure (Nilsson, 1971) taking into account the specific character of the defined graph (the interpretation of nodes as symptoms; some of them can be known *a priori* as true/false) and its elements (i.e. the AND, OR and NOT connections). Thus, informally speaking, the problem of finding possible (potential) diagnoses is equivalent to finding a (minimal) set of initial symptoms defining faulty components, control actions and external signals, such that a combination of initial causes implies the observed abnormality. In fact, a *diagnosis* may be any consistent combination of values of symptoms from D, justifying the observed abnormal behaviour with respect to the graph structure and consistent with observations.

The search for potential diagnoses is performed in a backward manner. For a given set of manifestations one has to hypothesize and explore a possible status of former nodes implying the observed status of manifestations. Let us recall that any node can take one of the following states: be known (or assumed) to be *true*, be known (or assumed) to be *false* or be of unknown status. The current assignment of truth-values to symptoms in the graph (possibly not to all of them) determines the current state of the search.

Formally, in order to define the state of the search, we introduce a mapping of the form

$$N \longrightarrow \{true, false, unknown\}$$

assigning to any symptom its current status. In practice, the initial state of the search is defined by a specification of the manifestations true and false (M^+ and M^-) and the observations of true and false symptoms (N^+ and N^-). During the search some new nodes are supposed to be true or false. Thus, at any stage of the search, the set of nodes *true* is of the form $S^+ = M^+ \cup N^+ \cup N'^+$ and the set of nodes *false* is of the form $S^- = M^- \cup N^- \cup N'^-$.

Symptoms of the value *unknown* are not represented explicitly in the state set. Further on, it is assumed that any state defined by S^+ and S^- is consistent, i.e. $S^+ \cap S^- = \emptyset$—the intersection of true and false symptoms is empty. Moreover, we shall also assume that any state set is maximal with respect to the possibility of information propagation. Whenever some new node is evaluated or assumed to be true/false, this information influences the state of the search. The rules of propagation are defined below.

Definition 3. The following points define propagation of information in AND/OR/NOT causal graphs:

Forward propagation: (causality, a simple logical interpretation assumed)

- *OR node true*: if at least one of the predecessors of an OR node is *true*, then the value of the OR node is set to *true*,
- *AND node false*: if at least one of the predecessors of an AND node is *false*, then the value of the AND node is set to *false*.
- *NOT node true*: if a predecessor of a NOT node is *false*, then the value of the NOT node is set to *true*,
- *NOT node false*: if a predecessor of a NOT node is *true*, then the value of the NOT node is set to *false*.

Forward propagation: (causality, a simple logical interpretation assumed; furthermore, the predecessors of a node are assumed to be all the direct causes for it)

- *OR node false*: if all the predecessors of an OR node are *false*, then the value of this OR node is set to *false*,
- *AND node true*: if all the predecessors of an AND node are *true*, then the value of this AND node is set to *true*.

Backward propagation: (causality, a simple logical interpretation assumed)

- *OR node false*: if an OR node is *false*, then the values of all its predecessors are set to *false*,
- *AND node true*: if an AND node is *true*, then the values of all its predecessors are set to *true*,
- *NOT node true*: if a NOT node is *true*, then the value of its predecessor is set to *false*,
- *NOT node false*: if a NOT node is *false*, then the value of its predecessor is set to *true*.

Backward propagation: (causality, a simple logical interpretation assumed; furthermore, the predecessors of a node are assumed to be all the direct causes for it)

- *OR node true*: if an OR node is *true*, then at least one of its predecessors must be assumed to be *true*; since in case of more than one predecessor the node selection is indeterministic, this rule is used only in the backward search procedure,
- *AND node false*: if an AND node is *false*, then at least one of its predecessors must be assumed to be *false*; again, since the node selection is indeterministic, this rule is used only in the search procedure.

The above rules define the principles of state propagation. Whenever a rule is applicable, a new symptom value is generated. It is next placed in the set representing the current state. In case some symptom turns out to take two inconsistent values, the initial state for propagation is considered to be inconsistent and it is not taken into account any more. In practice, this has the effect of *failure* and *backtracking* in PROLOG.

Now, let D be any set of symptoms together with their values, i.e. $D = D^+ \cup D^-$, where D^+ specifies which symptoms are true, and D^- gives false symptoms. Furthermore, let $S = S^+ \cup S^-$ denote a state of the graph providing the set of symptoms true and false, respectively. The state specified by S is said to follow from (or be implied by) D iff it can be generated from D with respect to the above propagation rules. This will be denoted by $D \vdash S$.

An initial change in a node status is a result of failure detection. The appropriate nodes belonging to M change their status from *unknown* to *true* or *false*. Further changes of the current state of the graph can be performed as a result of the following operations:

- assumption, i.e. a hypothesis formation concerning some node performed by graph-search procedure; this change is always from *unknown* to *true/false*,
- performing a test, i.e. probing (or just observation); the test assigns values *true/false* to one or more nodes,

- finally, after any such change, the propagation rules are applied so that all possible changes are performed,
- last but not least, if after that the new state is inconsistent, some assumption about symptom status must be withdrawn (backtracking) so that consistency be regained.

With respect to the definition of an AND/OR/NOT causal graph and the diagnostic problem, a solution to the problem is any consistent combination of the values of initial causes, such that it implies the observed malfunctions. The maximal state implied by the diagnosis must be internally consistent, and it must be consistent with observations. The precise definition of a diagnosis is as follows:

Definition 4. Let (G, M^+, M^-, N^+, N^-) denote a diagnostic problem. A possible diagnosis constituting a solution to the diagnostic problem is any (minimal) set D of all the initial nodes true D^+ and false D^- $(D = D^+ \cup D^-)$ such that:

- $D^+ \cup D^- \vdash M^+ \cup M^-$, i.e. the diagnosis implies the observed manifestations,
- if $S = S^+ \cup S^-$ describe the maximal state implied by D, then $S^+ \cap S^- = \emptyset$, i.e. the maximal state following from the diagnosis must be consistent,
- $N^+ \cap S^- = \emptyset$ and $N^- \cap S^+ = \emptyset$, i.e. the implied state is consistent with the observations.

According to the above definition, a diagnosis is assumed to be minimal, i.e. no proper subset of it is a diagnosis. One can also consider non-minimal diagnoses, or diagnoses minimal with respect to an induced subgraph of G (Fuster Parra, 1996; Ligęza *et al.*, 1996).

An illustration of example solutions for the problem specified with the graph of Fig. 1 are shown (together with the induced graphs) in Fig. 2; the three graphs constitute some possible solutions for failure m_1. The bottom nodes constitute possible minimal diagnoses.

The search for a solution is performed by a backward search procedure selecting and hypothesizing the state of predecessor nodes so that the state of the problem defining nodes is explained. The following points specify the rules of backward search in the case of AND, OR and NOT connections used:

- *OR node true*: explained by selecting one of its predecessors and setting it to *true*,
- *AND node true*: explained only by setting all of its predecessors to *true*,
- *OR node false*: explained only by setting all of its predecessors to *false*,
- *AND node false*: explained by setting one of its predecessors to *false*,
- *NOT node true*: explained by its predecessor being *false*,
- *NOT node false*: explained by its predecessor being *true*,
- *initial nodes*: initial nodes are set to *true* or *false* when necessary; arriving at an initial node completes the search on the selected path.

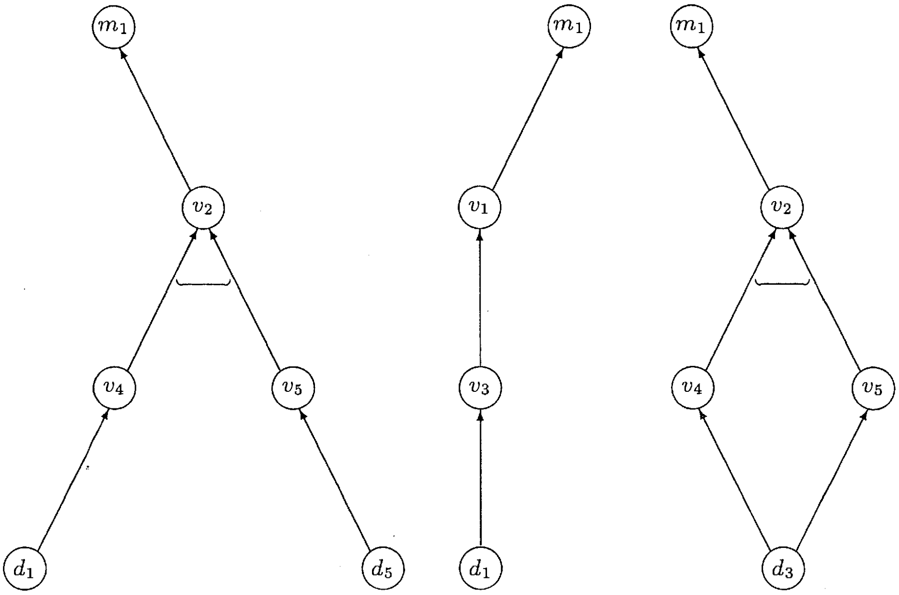

Fig. 2. Example solutions to a diagnostic problem.

The operations are performed only if they do not lead to inconsistency. The node selection procedure may be systematic, indeterministic, or heuristic. Since the algorithm always selects a single node in the case of OR node *true* and AND node *false*, the generated solutions are, in general, minimal.

Note that with respect to the definition, a solution to the diagnostic problem, i.e. a diagnosis, constitutes in fact a possible explanation of the observed failures. In order to verify it in practice, one should test all its elements. Validation of generated diagnoses is the next step of the diagnostic procedure.

5. Validation of Diagnoses

With respect to the model of diagnostic knowledge representation in the form of an AND/OR/NOT causal graph, any of the generated diagnoses constitutes a sufficient explanation of the observed faulty behaviour. However, usually only one of the diagnoses is the valid one, i.e. the one defining real troubles in the diagnosed system. In fact, *validation* of the obtained candidate diagnoses to identify the correct one should be performed. Validation can potentially be done by simulation of the influence of the hypothesized elementary faults on the observable behaviour of the system. However, this would require a more complete model of the system, covering abnormal behaviour, so that all but one of the potential diagnoses can be eliminated. This, however, is hardly possible in real cases. Thus the principle method consists mostly in testing the potential elementary causes in a direct or indirect manner.

For obvious reasons, minimal diagnoses are preferred. This point of view reflects the most natural tendency to find as simple explanations as possible (*principle of parsimony*, see (Reiter, 1987)). The problem of efficient validation of the final set of possible diagnoses is briefly considered below.

Let us assume that after a systematic search of the causal graph a set of possible diagnoses $D_1, D_2, \ldots, D_l$ is obtained. Any of these diagnoses is of the form $D_i^+ \cup D_i^- = \{d_i^1, d_i^2, \ldots, d_i^j, \bar{d_i}^{j+1}, \bar{d_i}^{j+2}, \ldots, \bar{d_i}^k\}$. Each D_i is assumed to be minimal and consistent, $i = 1, 2, \ldots, l$. In case consideration of minimal diagnosis is not sufficient, all superset diagnoses should be further considered.

A strategy for validation of diagnoses can be built on a number of assumptions and additional knowledge (e.g. the use of probabilities and minimization of entropy). Here a strategy based on a heuristic approach is proposed. The main idea is to separate the set of currently considered diagnoses into two almost equal, disjoint sets as a result of a test of one symptom, being a common element of the diagnoses.

In the sequel, the test is assumed to be an algorithmic procedure providing the truth value of a single symptom being an element of some diagnoses. The test can confirm a diagnosis if the tested symptom included in a diagnosis is found to have the status identical to its status in this diagnosis; otherwise, the test rejects the current diagnosis.

A diagnosis D is contradictory to the diagnosis D' if there exists an element d such that $d \in D$ and $\bar{d} \in D'$ or vice versa. Such an element will be referred to as a conflicting element with respect to diagnoses D and D'. Since the result of testing is unknown *a priori*, it is heuristically justified to test first conflicting elements—no matter what the result of the test is, at least one of the diagnoses is rejected. The "most useful" test is the one rejecting the maximal number of diagnoses at a time. The number of rejected diagnoses is, however, unknown *a priori*. Thus, the following heuristic approach based on "almost equal partitions" is put forward.

Let $n^+(d)$ denote the number of occurrences of d in the analyzed set of diagnoses $D_1, D_2, \ldots, D_l$. Moreover, let $n^-(d)$ denote the number of occurrences of $\bar{d}$ in the diagnoses. Thus, the least number of diagnoses rejected after a test of element d, denoted by $r(d)$, can be determined as

$$r(d) = \min\Big(n^+(d), n^-(d)\Big)$$

and the proposed selection of element(s) d to be tested should maximize $r(d)$.

Let d^* denote the symptom selected to be tested first. There should be

$$r(d^*) = \max_{d \in D_1, D_2, \ldots D_l} \Big(r(d)\Big)$$

In case the selection of d^* is not unique, another heuristic of cost criteria can be taken into account. Again, having sufficient knowledge about underlying probabilities, the above test can be generalized so as to select a test rejecting the maximal expected or average number of diagnoses.

6. Concluding Remarks

Let us briefly summarize the most important concepts presented in this paper. First, a generic and general model for the diagnostic procedure has been proposed and its initial stage of failure detection and classification based on the expected behaviour has been pointed out. The main idea concerning the diagnostic process is to consider it to be a multistage, multiple-approach sequential search procedure performed through hypothesis-test-ordering steps. Final validation of possible diagnoses obtained during the search has been discussed. The diagnoses themselves are not limited to faults of components. "Wrong" combinations of control actions and operational signals are also taken into account. A basic, core, and uniform causal structure in the form of an AND/OR/NOT causal graph, which incorporates basic logical concepts, is defined as a model of the search space for the diagnostic process.

A computer program, consisting mostly of a meta-interpreter of the rules given in this paper was implemented in PROLOG. The program was run on several test problems, including a simple tank system, a heating system of a house, the full-adder example (Reiter, 1987) and some other abstract problems. The detailed problem specification can be found in (Fuster Parra, 1996; Ligęza *et al.*, 1996). The results confirmed that a backward search on the causal graph structure can constitute an efficient tool for automatic generation of possible diagnoses. The main problem for practical applications consists in knowledge acquisition, i.e. building the appropriate graph. However, for certain systems (e.g. the ones composed of logical gates) this stage can be performed in a customary way (Ligęza *et al.*, 1996).

Further extensions: A number of relatively new ideas aimed at extensions of the basic model and enhancement of reasoning and search efficiency can be proposed and discussed in a more formal framework. Among other things, the ideas of the state of the search and test have been introduced and their use has been outlined in (Ligęza and Fuster Parra, 1994; 1995b). Some ideas concerning ordering of the search based on qualitative probabilities of faults have been proposed in (Fuster Parra and Ligęza, 1995b) and an extension of the notion of fault towards a "degree of faultiness" or "fuzzy faults" has been presented in (Fuster Parra and Ligęza, 1995a). Several further extensions concerning finite multiple-state models, constructive use of constraints for information propagation and validation strategy for possible diagnoses have been discussed in (Fuster Parra, 1996; Ligęza and Fuster Parra, 1994; Ligęza *et al.*, 1996).

Related work: All the presented ideas make direct use of the man-applied heuristic approach and remain in close relation with the engineering practice. The incorporation of them in more sophisticated diagnostic support systems is aimed at improving the efficiency of search for a final diagnosis and extending the basic model over more complex phenomena.

The present approach can be regarded as an extension and new application of fault trees used in the engineering practice for fault analysis and safety assignment (Barlow and Lambert, 1975). It extends the notion to graphs incorporating the basic logical connectives and their use as search space defining structures. Of course, the graphs do not need to be defined explicitly. A multiple use of prespecified component descriptions and recursive definitions are possible.

In AI, the work is mostly related to the abductive approaches (Poole, 1989) based on causal reasoning. A most comprehensive approach to the diagnosis based on causal graphs seems to be the one represented by Console *et al.* (Console and Torasso, 1988; 1992; Console *et al.*, 1989; Torasso and Console, 1989) which inspired the present research in many points. The main differences consist in a different definition and use of a causal graph. The graph defined here, refined and uniform in structure, serves as a direct tool for a *search* (backwards) for diagnoses while in the above-mentioned approach a causal graph is mostly used for construction of a logical model for consistency-based reasoning. Furthermore, the present approach is aimed at extending the area of applications and improving the efficiency with a proper use of tests and qualitative ordering of the search.

When taking the work on set covering (Reggia *et al.*, 1983; 1985), the proposed model can be regarded as a significant extension, consisting in introducing a logical structure lying between the sets of manifestations $\boldsymbol{M}$ and elementary diagnoses $\boldsymbol{D}$ (diseases in (Reggia *et al.*,1983)). Although the present approach was developed independently from that work, an attempt to keepi a similar notation was made in order to underline the common points.

In the area of automatic control, the present approach can be related to that of (Kościelny, 1995a; 1995b; Kościelny and Pieniążek, 1994). The main extensions are similar to (Reggia *et al.*, 1983; 1985) and they consist in introducing an extended structure of the causal graph versus the diagnostic relation used in the above-mentioned works.

A recent work on the diagnosis using explicit means-end models (Larsson, 1996) seems to support the points advocated in this work. The most principal common idea, although quite general, is that certain models can be used directly for the diagnosis understood as a search for the causes of faults. In the case of certain well-defined components, such as logical gates, there exists a simple methodology for almost direct encoding of the structure of an AND/OR/NOT causal graph based on the functional structure of the analyzed system (Ligęza *et al.*, 1996).

Several other works weakly related to the proposed approach may also be found, e.g. (Lunze and Schiller, 1992; Togueni *et al.*, 1993), which served as sources for auxiliary inspiration.

Acknowledgement

The research was supported by the State Committe for Scientific Research (grant No. 8 T11A 013 08) in the case of the first Author. The authors wish to thank the anonymous Reviewers for their comments improving the quality of this paper.

References

Barlow R.E. and Lambert H.E. (1975): *Introduction to fault tree analysis*, In: Reliability and Fault Tree Analysis. Theoretical and Applied Aspects of System Reliability and Safety Assessment (Barlow R.E., Fussel J.B. and Singpurwalla N.D., Eds.). — SIAM, Philadelphia, pp.7–35.

Chang S.J., DiCesare F. and Goldbogen G. (1991): *Failure propagation trees for diagnosis in manufacturing systems.* — IEEE Trans. Syst., Man, and Cybern., Vol.SMC–21, No.4, pp.767–776.

Console L. and Torasso P. (1988): *A logical approach to deal with incomplete causal models in diagnostic problem solving*, In: Uncertainty and Intelligent Systems, (Bouchon B. *et al.*, Ed.). — Berlin: Springer-Verlag, Lecture Notes in Computer Science, Vol.313, pp.255-264.

Console L. and Torasso P. (1992): *An approach to the compilation of operational knowledge from causal models.* — IEEE Trans. Syst., Man, and Cybernetics, Vol.SMC–22, No.4, pp.772–789.

Console L., Dupré D.T. and Torraso P. (1989): *A theory of diagnosis for incomplete causal models.* — Proc. IJCAI'89, Detroit, pp.1311–1317.

Davis R. (1984): *Diagnostic reasoning based on structure and behavior.* — Artificial Intelligence, Vol.24, pp.347–410.

Davis R. (1993): *Retrospective on diagnostic reasoning based on structure and behavior.* — Artificial Intelligence, Vol.59, No.1–2, pp.149–157.

Davis R. and Hamscher W. (1992): *Introduction to model-based diagnosis*, In: Readings in Model-Based Diagnosis, (Hamscher W., Console L. and DeKleer J., Eds.), San Mateo, CA: Morgan Kaufmann Publ., pp.3–24.

DeKleer J. and Williams B.C. (1987): *Diagnosing multiple faults.* — Artificial Intelligence, Vol.32, No.1, pp.97–130.

Frank P.M. and Köppen-Seliger B. (1995): *New developments using AI in fault diagnosis.* — Prep. IFAC Int. Workshop *Artificial Intelligence in Real-Time Control*, Bled, Slovenia, pp.1–12.

Fuster Parra P. (1996): *A Model for Causal Diagnostic Reasoning. Extended Inference Modes and Efficiency Problems.* — Ph.D. Thesis, University of Balearic Islands.

Fuster Parra P. and Ligęza A. (1995a): *Fuzzy fault evaluation in causal diagnostic reasoning*, In: Applications of Artificial Intelligence in Engineering X, (Adey R.A. *et al.*, Eds.). — Boston: Computational Mechanics Publ., pp.137–144.

Fuster Parra P. and Ligęza A. (1995b): *Qualitative probabilities for ordering diagnostic reasoning in causal graphs*, In: Research and Development in Expert Systems XII, (Bramer M.A. *et al.*, Eds.). — Oxford: SGES Publications, Information Press Ltd., pp.327–339.

Fuster Parra P. and Ligęza A. (1996): *A model for representing causal diagnostic reasoning.* — Proc. 13th European Meeting *Cybernetics and Systems Research*, Vienna, Vol.2, pp.1211–1216.

Genesereth M.R. (1984): *The use of design descriptions in automated diagnosis.* — Artificial Intelligence, Vol.24, No.1–3, pp.411–436.

Isermann R. (1993): *On the applicability of model based fault detection for technical processes.* — IFAC World Congress, Sydney, Vol.9, pp.195–200.

Isermann R. (1994): *Integration of fault detection and diagnosis methods.* — Proc. SAFEPROCESS'94, Espoo, Finland, Vol.II, pp.597–612.

Korbicz J. (Ed.) (1995): *Methods and Techniques of Technical Diagnostics.* — Proc. Workshop, Zielona Góra: Lubusky Scientific Society, (in Polish).

Korbicz J. and Cempel C. (Eds.) (1993): *Analytical and Knowledge-Based Redundancy in Fault Detection and Diagnosis.* — Appl. Math. and Comp. Sci., Special Issue, Vol.3, No.3.

Korbicz J., Obuchowicz A. and Uciński D. (1994): *Artificial Neural Networks. Foundations and Applications.* — Warszawa: Akademicka Oficyna Wydawnicza PLJ, (in Polish).

Kościelny J.M. (1995a): *Fault isolation in industrial processes by the dynamic table of states method.* — Automatica, Vol.31, No.5, pp.747–753.

Kościelny J.M. (1995b): *Rules of fault isolation.* — Archives of Control Sciences, Vol.4 (XL), No.3–4, pp.321–336.

Kościelny J.M. and Pieniążek A.M. (1994): *Algorithm of fault detection and isolation applied for an evaporate unit in a sugar factory.* — Control Eng. Practice, Vol.2, No.4, pp.649–657.

Larsson J.E. (1996): *Diagnosis based on explicit means-end models.* — Artificial Intelligence, Vol.80, No.1, pp.29–93.

Ligęza A. (1995): *A model for diagnostic inference. Knowledge representation and processing.* — Research Report Institute of Automatics AGH, No.54, Cracow.

Ligęza A. and Fuster Parra P. (1994): *Qualitative knowledge representation and processing for causal diagnostic reasoning. Extended reasoning modes and efficiency-related issues.* — Research Report of the Institute of Automatics AGH, No.38, Cracow.

Ligęza A. and Fuster Parra P. (1995a): *Automated diagnosis: an expected-behaviour based approach.* — Proc. Conf. *System, Modelling, Control'95*, Zakopane, Poland, Vol.2, pp.7–12.

Ligęza A. and Fuster Parra P. (1995b): *An approach to diagnosis through search of AND/OR/NOT causal graphs.* — Prep. IFAC/IMACS Int. Workshop *Artificial Intelligence in Real-Time Control*, Bled, Slovenia, pp.126–131.

Ligęza A., Fuster Parra P. and Aguilar-Martin J. (1996): *Causal abduction: Backward search on causal logical graphs as a model for diagnostic reasoning.* — LAAS du CNRS Report, No.96316, Toulouse.

Lunze J. and Schiller F. (1992): *Logic-based diagnosis utilizing the causal structure of dynamical systems.* — Prep. IFAC/IFIP/IMACS Int. Symp. *Artificial Intelligence in Real-Time Control*, Delft, pp.649–654.

Montmain J. and Leyval L. (1994): *Causal graphs for model based diagnosis.* — Proc. SAFEPROCESS'94, Espoo, Finland, pp.347–355.

Nilsson N.J. (1971): *Problem-Solving Methods in Artificial Intelligence.* — New York: McGraw-Hill.

Poole D. (1989): *Normality and faults in logic based diagnosis.* — Proc. IJCAI'89, Detroit, pp.1304–1310.

Reiter R. (1987): *A theory of diagnosis from first principles.* — Artificial Intelligence, Vol.32, No.1, pp.57–95.

Reggia J.A., Nau D.S. and Wang P.Y. (1983): *Diagnostic expert system based on set covering model.* — Int. J. Man-Machine Studies, Vol.19, pp.437–460.

Reggia J.A., Nau D.S. and Wang P.Y. (1985): *A formal model of diagnostic inference. I. Problem formulation and decomposition.* — Information Sciences, Vol.37, pp.227–256.

Saucier G., Ambler A. and Breuer M.A. (Eds.) (1989): *Knowledge Based Systems for Test and Diagnosis.* — Amsterdam, New York: North-Holland.

Struss P. (1992): *Knowledge-based diagnosis – An important challenge and touchstone for AI.* — Proc. ECAI'92, Wien, John Wiley and Sons, Ltd., pp.863–874.

Torasso P. and Console L. (1989): *Diagnostic Problem Solving. Combining Heuristic Approximate and Causal Reasoning.* — London: North Oxford Academic.

Togueni A.K.A., Craye E. and Gentina J.C. (1993): *An approach for the placement of sensors for on-line diagnostic purposes.* — Proc. IFAC World Congress, Sydney, Vol.8, pp.83–88.

Tzafestas S.G. (Ed.) (1989): *Knowledge-Based System Diagnosis, Supervision and Control.* — New York, London: Plenum Press.

Received: February 1, 1996
Revised: November 29, 1996