Diagnosability analysis without fault models [1]

Xavier Pucel * **Wolfgang Mayer** * **Markus Stumptner** *

* *University of South Australia, Advanced Computing Research Center,
5095 Mawson Lakes SA, Adelaide, Australia
(e-mail: FirstName.LastName@unisa.edu.au)*

Abstract: This paper addresses the problem of diagnosability analysis, which allows a system designer to anticipate the performance of a model based diagnosis (MBD) algorithm for a given system. Such analysis requires a formal description of the system behavior, called model, which can be very difficult to establish, especially when faults occur in the system. Despite this, all known diagnosability frameworks rely on some specification of the system behavior under the absence and presence of faults.
This paper presents a diagnosability analysis algorithm related to a diagnosis approach in which the model of faulty system components is unspecified. Diagnosis is based on the description of the normal behavior as well as a decomposition of the system into components, and assesses which components cannot be behaving normally. Diagnosability is defined in a way that copes with the capabilities of such diagnosis approaches. An algorithm for checking diagnosability incrementally or hierarchically is described and illustrated.

Keywords: Model-Based Diagnosis, Diagnosability analysis, Software debugging

1. INTRODUCTION

Model-Based Diagnosis has received an increasing interest during recent years, and has been successfully applied many times. Experience has proved that diagnosis provides better results when taken into account as early as the system design stage. The problem of estimating at design time the performance that a given diagnosis algorithm will provide at run time is known as diagnosability analysis. This particular problem of diagnosability analysis has received a significant interest from the model based diagnosis community, in particular by Dressler and Struss [2003], Sampath et al. [1995], Travé-Massuyès et al. [2006], Pencolé and Cordier [2005], Cordier et al. [2006], yet all known diagnosability analysis approaches rely on some specification of the system behavior under the presence of faults. This requirement is particularly difficult to fulfill, since fault models are often difficult to establish in real systems. As a consequence, diagnosis approaches relying only on normal behavior model are commonly used, although diagnosability analysis is impossible in such a situation.

Diagnosis approaches using only a normal behavior model are among the most classical of their fields. In the FDI community, the constraints that define the normal behavior are derived into a set of consistency tests, that are informally associated to different system faults. In DX approaches, the system is decomposed into components, and diagnosis traditionally identifies the smallest sets of components that cannot all behave normally, named minimal conflict sets. Cordier et al. [2004] proved that by defining components in FDI approaches and associating the consistency tests to sets of components, both approaches could lead to the same results. Consequently, the efficient constraint combination techniques from FDI can be used to build a complete list of potential minimal conflicts and diagnoses. This work provides essential grounds for diagnosability analysis in the absence of fault models.

Some diagnosis approaches indirectly specify fault models. In particular, FDI approaches that rely on a fault signature matrix strongly associate a fault to the violation of some constraints on observable variables. This is equivalent to constraining the faulty behavior with the negation of these constraints. In some DX approaches, the component exoneration assumption states that a faulty component necessarily exhibits a different behavior from its normal behavior. This is equivalent to constraining the faulty behavior by the negation of the normal behavior constraints. These models can be addressed with existing diagnosability approaches, and are not considered in this paper.

This paper establishes definitions that allow to characterize the diagnosability of systems in the absence of fault models. We argue why existing approaches are inadequate when applied directly under these assumptions, and adapt definitions to suit the modified context. The main aspect of the adaptation is that when the faulty behavior is not specified, the number of diagnosis candidates gets very large, and the diagnosis process arbitrarily eliminates some unlikely candidates for the sake of tractability and usability; the most common way to do so is to eliminate non-minimal diagnoses. A definition of discriminability of two combinations of faults is established, and diagnosability is characterized by the set of all discriminability results. An algorithm is described, that allows to analyze

[1] This work was supported by the Australian Research Council under grant DP0881854.

the diagnosability of subsystems and aggregate the results in order to obtain diagnosability for the whole system. This algorithm offers early detection of non-diagnosable fault combinations, and provides a great flexibility in the choice of subsystems and merging sequence. Analysis can be performed incrementally or hierarchically, according to the user preference or to the system's natural structure.

This paper is organized as follows: first the diagnosis approach, inspired from Hamscher et al. [1992], is recalled is section 2, then definitions for diagnosability adapted for our approach are given in section 3. An incremental algorithm is described and illustrated in section 4. Comparison with other diagnosability analysis approaches and related work is discussed in section 5.

2. MODEL-BASED DIAGNOSIS

The diagnosis approach considered in this paper is the so-called *conflict based approach* [Hamscher et al., 1992], of which we recall the most important definitions. A system is represented by a finite set V of variables, each variable ranging over a finite domain. The system is decomposed in a set COMPS of components, each component $c_i \in$ COMPS is associated with a model written in first order logic. Faults are modeled by the predicate ABmeaning "abnormal": $\mathrm{AB}(c_i)$ means that a fault has occurred in component c_i. The system description SD is assumed to be expressed in the following form:

$$\mathrm{SD} \equiv \bigwedge_{c_i \in \mathrm{COMPS}} \neg\mathrm{AB}(c_i) \Rightarrow \mathit{Model}(c_i), \tag{1}$$

where $\mathit{Model}(c_i)$ is a conjunction of first order logic sentences constraining the values of some variables in V. These variables are constrained only when the component c_i is behaving normally (that is, $\neg\mathrm{AB}(c_i)$ holds). When $\mathrm{AB}(c_i)$ holds, these variables range unrestricted over their respective domains.

The interaction between components is represented by shared variables: let $\mathit{sco}(M_i)$ denote the set of variables constrained by M_i. Two components c_i and c_j are connected if and only if $\mathit{sco}(M_i) \cap \mathit{sco}(M_j)$ is not empty.

A diagnosis is represented by the sets of suspected components. More precisely, for a set of components $\Delta \subseteq$ COMPS, let:

$$D(\Delta) \equiv \bigwedge_{c_i \in \Delta} (\mathrm{AB}(c_i)) \wedge \bigwedge_{c_i \in \mathrm{COMPS}\setminus\Delta} (\neg\mathrm{AB}(c_i)) \tag{2}$$

Let *obs* be an assignment to some variables representing an observation. Then Δ is a *diagnosis* if and only if:

$$\mathrm{SD} \wedge \mathit{obs} \wedge D(\Delta) \text{ is satisfiable}$$

Since components assumed to be faulty do not constrain the values of variables in the system model, any super-set Δ' of Δ is also a diagnosis. A set of components Δ is a *minimal diagnosis* if and only if Δ is a diagnosis and every $\Delta' \subset \Delta$ is not a diagnosis. Most algorithms aim at finding *minimal diagnoses.*

3. DIAGNOSABILITY

Although there is great diversity among existing diagnosability definitions and algorithms, most rely on the same principles and can be expressed under a common framework [Cordier et al., 2006]. In particular, all known approaches to diagnosability analysis rely on the specification of the system's normal behavior, as well as its behavior when faults occur. In such approaches, fault candidates can be eliminated if assuming their presence contradicts the observation. Diagnosability analysis aims at finding observations that are consistent with several combinations of faults. If such observations exist, then the system is not diagnosable, as some combinations of faults may not be distinguishable from others given the observations available in the system.

In our approach, such reasoning is impossible because faulty behavior is not specified. When faulty, a component may adopt any behavior, including its normal behavior. As a consequence, it is impossible to discard a fault hypothesis because it is inconsistent with the observations. In particular, COMPS is a diagnosis for any observation. Diagnosis approaches deal with this issue by looking for minimal diagnoses, discard faults when their absence is consistent with the observation. This requires to adapt well-known notions of diagnosability to suit the consistency based diagnosis framework.

In the remaining section, we adapt diagnosability definitions to suit the consistency-based framework. First, the notion of diagnosability is introduced and formalized. We then present a modified framework for diagnosability that does not require strong component fault models and explore its characteristics.

3.1 General concepts

Various concepts used in diagnosability analysis are still useful in this approach. In particular, the concepts of fault mode and observable are directly adaptable. A *fault mode* is a behavioral mode of the system that is associated with the presence of some faults and the absence of the other faults. It can be represented by the set containing the faults present in the system. The normal mode is the fault mode that corresponds to the absence of all faults, and represented by $\emptyset$. The set of all fault modes is denoted $\mathcal{F}$.

Formally, a fault mode is represented as a set of faults, which, in our approach, is equivalent to a set of components. Although a fault mode appears to be similar to a diagnosis, they should not be interpreted in the same way. A diagnosis is an explanation of a given observation, while a fault mode is a behavioral mode of the system that may be associated with observables. Consequently, $\mathcal{F} = 2^{\mathrm{COMPS}}$. When the system is in fault mode f, its behavior is represented by the constraint $\mathrm{SD} \wedge D(f)$.

Some variables of the model SD are considered to be observable, which means that their value is known when the system is running. It is assumed that every execution of the system leads to an observation represented by a value tuple (*"observable"*) for the observable variables. The set of all reachable value tuples is called the *set of observables* and denoted OBS.

Fault modes are related to observables via a signature function:

$$\mathit{Sig} : \mathcal{F} \rightarrow 2^{\mathrm{OBS}}$$

The *signature* of a fault mode f is the set of observables that are reachable when the system is in mode f.

The set of observable variables is denoted V_{OBS}. The diagnosability approach relies on a projection operation $\mathcal{P}_{\mathrm{OBS}}$, that projects constraints on observable variables. As a constraint describes a set of system states, its projection describes the corresponding set of observations. Formally, if C is a constraint, the projection on observable variables $\mathcal{P}_{\mathrm{OBS}}(C)$ is the smallest constraint with $sco(\mathcal{P}_{\mathrm{OBS}}(C)) = V_{\mathrm{OBS}}$ such that if an assignment γ to all the system variables satisfies a constraint C then the restriction of γ to observable variables satisfies $\mathcal{P}_{\mathrm{OBS}}(C)$.

Consequently OBS is the set of value tuples for observable variables satisfying $\mathcal{P}_{\mathrm{OBS}}(\mathrm{SD})$. Moreover the signature of a fault mode can then be defined as follows:

$$\begin{aligned}&\forall v_o \in \mathrm{OBS}, \forall f \subseteq \mathrm{COMPS},\\ &\qquad v_o \in Sig(f) \Leftrightarrow v_o \text{ satisfies } \mathcal{P}_{\mathrm{OBS}}\big(\mathrm{SD} \wedge D(f)\big)\end{aligned} \tag{3}$$

3.2 Signature lattice

The previous definitions hold for every diagnosability approach. When faulty behavior is specified, one can hope that for two different fault modes f_1 and f_2, the system behaviors $\mathrm{SD} \wedge D(f_1)$ and $\mathrm{SD} \wedge D(f_2)$ are different enough to lead to different observations, hence $Sig(f_1) \cap Sig(f_2) = \emptyset$. There is no a priori reason for two fault modes to share common behaviors.

In our case, the behavior of a faulty component is not specified, which means that the component may adopt any behavior when faulty. As a consequence, as faults appear, the constraints defining the system behavior simply loosen. More precisely, for any fault mode f, we have (by equations (1) and (2)):

$$\mathrm{SD} \wedge D(f) \equiv \bigwedge_{c_i \in \mathrm{COMPS} \setminus f} Model(c_i)$$

The behavior of the system under a fault mode f is defined as the conjunction of the constraints associated with the components that behave normally. As a consequence, for all fault modes f_1 and f_2,

$$f_1 \subset f_2 \Rightarrow \Big(\big(\mathrm{SD} \wedge D(f_1)\big) \Rightarrow \big(\mathrm{SD} \wedge D(f_2)\big)\Big)$$

holds. The implication $A \Rightarrow B$ means that the set of tuples representing observables satisfying A is a subset of the tuples satisfying B. It follows that

$$f_1 \subset f_2 \Rightarrow Sig(f_1) \subseteq Sig(f_2) \tag{4}$$

All signatures are partially ordered by set inclusion, and form a complete lattice with $Sig(\emptyset)$ as lower bound, and $Sig(\mathrm{COMPS})$ as upper bound.

This result is significant, since it implies that all signatures are super-sets of $Sig(\emptyset)$, and no two signatures are disjoint. The definition of diagnosability given in Pucel [2008] cannot apply in our context.

In order to suit our context, we need to take into account that the diagnosis process does not consider all diagnoses, but only minimal diagnoses. Indeed, if the signature of a fault mode f contains an observable o, this means that when the system produces the observation o, f is a diagnosis. However, this does not guarantee that f is a minimal diagnosis for o.

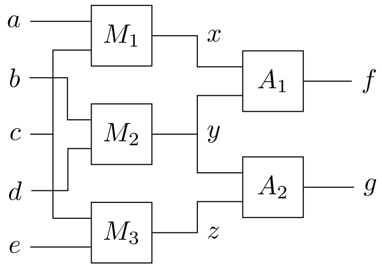

Fig. 1. A simple example composed of three multipliers and two adders.

3.3 Example

To illustrate the concepts above, let us introduce an old fashioned yet illustrative example. The system represented in figure 1 is modeled as follows:

$$V = \{a, b, c, d, e, x, y, z, f, g\}$$
$$\mathrm{COMPS} = \{M_1, M_2, M_3, A_1, A_2\}$$
$$\begin{aligned}\mathrm{SD} \equiv\ & \neg\mathrm{AB}(M_1) \Rightarrow (x = a \cdot c)\\ \wedge\ & \neg\mathrm{AB}(M_2) \Rightarrow (y = b \cdot d)\\ \wedge\ & \neg\mathrm{AB}(M_3) \Rightarrow (z = c \cdot e)\\ \wedge\ & \neg\mathrm{AB}(A_1) \Rightarrow (f = x + y)\\ \wedge\ & \neg\mathrm{AB}(A_2) \Rightarrow (g = y + z)\end{aligned}$$

In this system, all variables are integers, and addition and multiplication are standard arithmetic operations. We suppose that only input and output variables are observable, i.e. $V_{\mathrm{OBS}} = \{a, b, c, d, e, f, g\}$. Let us build the model for the normal mode:

$$\begin{aligned}\mathrm{SD} \wedge D(\emptyset) &\equiv \mathrm{SD} \wedge \neg\mathrm{AB}(M_1) \wedge \neg\mathrm{AB}(M_2)\\ &\quad \wedge \neg\mathrm{AB}(M_3) \wedge \neg\mathrm{AB}(A_1) \wedge \neg\mathrm{AB}(A_2)\\ &\equiv (x = a \cdot c) \wedge (y = b \cdot d) \wedge (z = c \cdot e)\\ &\quad \wedge (f = x + y) \wedge (g = y + z)\end{aligned}$$

By projecting on observable variables and considering the set of solutions, we get:

$$\begin{aligned}Sig(\emptyset) = \{&(a, b, c, d, e, f, g) |\\ &(f = a \cdot c + b \cdot d) \wedge (g = b \cdot d + c \cdot e)\}\end{aligned}$$

The signature of the normal mode is the smallest one. Indeed, by introducing faults in the system, we relax the constraints on the variables, allowing them to range more freely over their respective domains. For example, let us consider fault mode $\{M_1\}$, for which component M_1 is faulty while other components are normal. We have:

$$\begin{aligned}\mathrm{SD} \wedge D(\{M_1\}) &\equiv (y = b \cdot d) \wedge (z = c \cdot e)\\ &\quad \wedge (f = x + y) \wedge (g = y + z)\\ Sig(\{M_1\}) &= \{(a, b, c, d, e, f, g) | \quad g = b \cdot d + c \cdot e\}\end{aligned}$$

By applying the same reasoning to all the fault modes, we find that:

$$\begin{aligned}Sig(\{M_1\}) &= Sig(\{A_1\}) = Sig(\{A_1, M_1\})\\ &= \{(a, b, c, d, e, f, g) | \quad g = b \cdot d + c \cdot e\}\end{aligned}$$

$$\begin{aligned}Sig(\{M_3\}) &= Sig(\{A_2\}) = Sig(\{A_2, M_3\})\\ &= \{(a, b, c, d, e, f, g) | \quad f = a \cdot c + b \cdot d\}\end{aligned}$$

$$Sig(\{M_2\}) = \{(a, b, c, d, e, f, g) | \quad f - a \cdot c = g - c \cdot e\}$$

In all the other fault modes, observable variables are not constrained and range over their full integer domains. The signatures form a lattice as illustrated in figure 2.

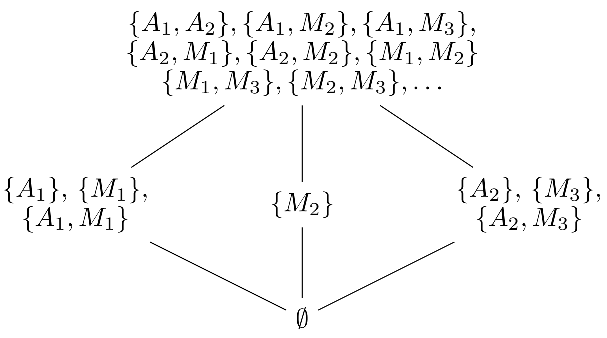
$\{A_1, A_2\}, \{A_1, M_2\}, \{A_1, M_3\},$
$\{A_2, M_1\}, \{A_2, M_2\}, \{M_1, M_2\}$
$\{M_1, M_3\}, \{M_2, M_3\}, \ldots$
$\{A_1\}, \{M_1\},$
$\{A_1, M_1\}$
$\{M_2\}$
$\{A_2\}, \{M_3\},$
$\{A_2, M_3\}$
$\emptyset$

Fig. 2. Fault modes of the system partially ordered by inclusion of their signatures. Fault modes with identical signatures are assigned to the same node. Fault modes involving three components or more belong to the upper bound and are not enumerated.

This lattice is to be interpreted as follows: with this system and some observation *obs*, whenever a fault mode f is a diagnosis, then every fault mode in the same lattice node and in every upper node is also a diagnosis.

The lattice of signatures can also be used to deduce information about minimal diagnoses. For example, it is easy to deduce that $\{A_1, M_1\}$ *cannot be a minimal diagnosis*, since when it is a diagnosis, then $\{A_1\}$ and $\{M_1\}$ also are diagnoses. This result is particularly interesting: we know that a diagnosis algorithm will never output $\{A_1, M_1\}$ as a minimal diagnosis for this system, for any observation. The same result holds for $\{A_2, M_3\}$ and for all fault modes containing three or more faults.

3.4 Diagnosability definitions

The previous example illustrates that although the faulty behavior of components is not specified, it is still possible to compute at design time important information about the diagnosis capabilities. The concept of signature provides information about the situations in which a fault mode will be a diagnosis, which is sufficient in classical diagnosability approaches. However, when we are concerned about minimal diagnoses, additional reasoning is needed.

This section introduces the concept of *minimal signature*, that describes the situations in which a fault mode is a minimal diagnosis. The minimal signature function associates each fault mode f to the set of observables for which f would be a minimal diagnosis. It can be defined as follows:

$$MinSig(f) = Sig(f) \setminus \bigcup_{f' \subset f} Sig(f') \tag{5}$$

This definition expresses the reasoning that f is a minimal diagnosis for the observables for which f is a diagnosis minus the observables for which some $f' \subset f$ is a diagnosis.

In the formalism used, a minimal signature can be computed directly from the model, by considering observables that satisfy $\mathcal{P}_{\mathrm{OBS}}\big(\mathrm{SD} \wedge D(f)\big)$ for the considered fault mode f, and that satisfy $\neg\mathcal{P}_{\mathrm{OBS}}\big(\mathrm{SD} \wedge D(f')\big)$ for smaller fault modes $f' \subset f$.

$$\forall v_o \in \mathrm{OBS}, \forall f \subseteq \mathrm{COMPS},$$
$$v_o \in MinSig(f) \Leftrightarrow v_o \text{ satisfies}$$
$$\mathcal{P}_{\mathrm{OBS}}\big(\mathrm{SD} \wedge D(f)\big) \wedge \bigwedge_{f' \subset f} \neg\mathcal{P}_{\mathrm{OBS}}\big(\mathrm{SD} \wedge D(f')\big) \tag{6}$$

This property can be helpful since it allows to check whether a given fault mode has a non-empty minimal signature without actually building it. The same principle can be applied to check whether two minimal signatures intersect.

The designer of a system is generally not interested in the contents of minimal signatures, only in specific properties of the sets:

$MinSig(f) = \emptyset$ means that f cannot be a minimal diagnosis. Even if the system really is in this fault mode, the diagnosis algorithm will output smaller minimal diagnoses. f is *not diagnosable.*

$MinSig(f_1) = MinSig(f_2)$ then f_1 is a minimal diagnosis if and only if f_2 is. f_1 and f_2 are *not discriminable.*

$MinSig(f_1) \subseteq MinSig(f_2)$ with $f_1 \not\subset f_2$, then whenever f_1 is a minimal diagnosis, so is f_2. f_2 is *weakly discriminable* from f_1, and f_1 is not discriminable from f_2.

$MinSig(f_1) \cap MinSig(f_2) = \emptyset$ means that f_1 and f_2 cannot be both minimal diagnoses at the same time (i.e., for the same observation). f_1 and f_2 are mutually *discriminable.*

The definitions of non-discriminability, weak discriminability and (strong) discriminability are inspired by Travé-Massuyès et al. [2006], although not equivalent (see section 5 for details).

These properties characterize diagnosability for diagnosis approaches where faulty behavior is not specified. Establishing diagnosability early in the system design phase is of particular significance if it is required that a certain level of diagnosability must be achieved for (a sub-set) of all components.

4. INCREMENTAL ANALYSIS

This section presents an incremental algorithm for computing minimal signatures, or more precisely the constraints of which minimal signatures contain the solutions. Small sets of components are analyzed separately, and the results of connected sets of components are aggregated one by one, discarding information at each operation.

An incremental approach is particularly helpful in the case of distributed systems, since diagnosability analysis can be performed on local sites before merging the local results in order to obtain an analysis for the whole system. In the case of modular systems, where a subsystem can be replaced or reused in another environment, the analysis can provide information about the diagnosability of this subsystem.

This algorithm first analyzes disjoint sets of components, and then aggregates the results until all the components have been considered by the analysis. There is no restriction on the order in which components should be analyzed, but the principle of incremental or hierarchical analysis suggests that the more strongly connected two components are, the sooner their combination should be considered. This is however not an intrinsic requirement in our framework, where components may be combined in any order.

4.1 Local analysis and merging

Local analysis requires the introduction of concepts to reason on only some components or subsystem of the system. The concepts of local model, local fault mode and interface variable are introduced in the following.

Let $S \subseteq \mathrm{COMPS}$ be a set of components to be analyzed locally. The model of the subsystem containing the components of S, also called *local model* and denoted SD_S is equal to:

$$\mathrm{SD}_S \equiv \bigwedge_{c\in S} \neg\mathrm{AB}(c) \Rightarrow \mathit{Model}(c) \tag{7}$$

Analysis of the local model only considers fault modes associated with components in S, formally represented by subsets of S. The translation of a local fault mode $f \subseteq S$ as a constraint is written as:

$$D_S(f) \equiv \bigwedge_{c_i\in f} \big(\mathrm{AB}(c_i)\big) \wedge \bigwedge_{c_i\in S\setminus f} \big(\neg\mathrm{AB}(c_i)\big) \tag{8}$$

For every local fault mode $f \subseteq S$ the behavior of the subsystem S is modeled by $\mathrm{SD}_S \wedge D_S(f)$.

Local analysis exploits interface variables in addition to observable variables to discriminate fault modes. The set of *interface variables* of S, denoted $\mathit{itf}(S)$, is defined as the set of variables that connect components in S to components outside of S:

$$\begin{aligned}\mathit{itf}(S) &= \quad \mathit{sco}(S) \quad \cap \quad \mathit{sco}(\mathrm{COMPS}\setminus S)\\ &= \Big(\bigcup_{c\in S} \mathit{sco}(c)\Big) \cap \Big(\bigcup_{c\in \mathrm{COMPS}\setminus S} \mathit{sco}(c)\Big)\end{aligned}$$

We define the set of *relevant variables* $\mathit{rel}(S) = \mathit{itf}(S) \cup (V_{\mathrm{OBS}} \cap \mathit{sco}(S))$ that contains interface and observable variables in S. Note that $\mathit{rel}(\mathrm{COMPS}) = V_{\mathrm{OBS}}$, since the whole system has no interface variables.

The projection operation on observable variables $\mathcal{P}_{\mathrm{OBS}}$ is extended for any target set of variables. For any set of variables *var*, $\mathcal{P}_{var}$ is defined as follows: an assignment γ of all variables in V satisfies a constraint C if and only if the restriction of γ to the variables in *var* satisfies $\mathcal{P}_{var}(C)$. We pay particular attention to the projection of local models on their relevant variables $\mathcal{P}_{\mathit{rel}(S)}\big(\mathrm{SD}_S \wedge D_S(f)\big)$.

These projected local models are then combined with the projection of other local models. Let S_1 and S_2 be two subsystems, disjoint ($S_1 \cap S_2 = \emptyset$) and connected ($\mathit{itf}(S_1) \cap \mathit{itf}(S_2) \neq \emptyset$). For every local fault modes $f_1 \subseteq S_1$ and $f_2 \subseteq S_2$, the behavior of the subsystem $S_1 \cup S_2$ under the local fault mode $f_1 \cup f_2$ has the following property:

$$\begin{aligned}&\mathrm{SD}_{S_1\cup S_2} \wedge D_{S_1\cup S_2}(f_1 \cup f_2) \equiv\\ &\qquad\qquad \big(\mathrm{SD}_{S_1} \wedge D_{S_1}(f_1)\big) \wedge \big(\mathrm{SD}_{S_2} \wedge D_{S_2}(f_2)\big)\end{aligned}$$

This is due to the fact that both SD_S and D_S are conjunctions of clauses as indicated in equations (7) and (8).

Moreover, since $\mathit{rel}(S_1)$ and $\mathit{rel}(S_2)$ both contain all the variables common to both S_1 and S_2 (i.e., $\mathit{sco}(S_1) \cap \mathit{sco}(S_2) \subset \mathit{rel}(S_i), i \in \{1, 2\}$), it is true that:

$$\begin{aligned}&\mathcal{P}_{\mathit{rel}(S_1\cup S_2)}\big(\mathrm{SD}_{S_1\cup S_2} \wedge D_{S_1\cup S_2}(f_1 \cup f_2)\big) \equiv\\ &\mathcal{P}_{\mathit{rel}(S_1\cup S_2)}\big(\mathrm{SD}_{S_1}\wedge D_{S_1}(f_1)\big)\wedge\mathcal{P}_{\mathit{rel}(S_1\cup S_2)}\big(\mathrm{SD}_{S_2}\wedge D_{S_2}(f_2)\big)\end{aligned} \tag{9}$$

Since $\mathit{rel}(\mathrm{COMPS}) = V_{\mathrm{OBS}}$, the combination of the local behavior models for all the system components establishes the signature of the combined fault mode.

4.2 Minimal signatures

As explained in section 3.4, we are not interested in signatures, or at least not as much as in minimal signatures. The minimal signatures can be built from the signatures constructed incrementally, a more efficient detection can be done at the local level.

During local analysis of a subsystem S, it is possible that two local fault modes $f_1, f_2 \subseteq S$ have equivalent projected behavioral models. In this case, let $S' = \mathrm{COMPS} \setminus S$ and f' be a local fault mode for S'. Then $f_1 \cup f'$ and $f_2 \cup f'$ are fault modes for the system, and their signatures contain the solutions for:

$$\begin{aligned}&\mathcal{P}_{\mathrm{OBS}}\big(\mathrm{SD} \wedge D(f_1 \cup f')\big) \equiv\\ &\qquad \mathcal{P}_{\mathit{rel}(S\cup S')}\big(\mathrm{SD}_{S\cup S'} \wedge D_{S\cup S'}(f_1 \cup f')\big) \equiv\\ &\quad \mathcal{P}_{\mathit{rel}(S)}\big(\mathrm{SD}_S \wedge D_S(f_1)\big) \wedge \mathcal{P}_{\mathit{rel}(S')}\big(\mathrm{SD}_{S'} \wedge D_{S'}(f')\big)\end{aligned}$$

And similarly for f_2:

$$\begin{aligned}&\mathcal{P}_{\mathrm{OBS}}\big(\mathrm{SD} \wedge D(f_2 \cup f')\big) \equiv\\ &\quad \mathcal{P}_{\mathit{rel}(S)}\big(\mathrm{SD}_S \wedge D_S(f_2)\big) \wedge \mathcal{P}_{\mathit{rel}(S')}\big(\mathrm{SD}_{S'} \wedge D_{S'}(f')\big)\end{aligned}$$

Consequently, since the projected local models $\mathcal{P}_{\mathit{rel}(S)}\big(\mathrm{SD}_S\wedge D_S(f_1)\big)$ and $\mathcal{P}_{\mathit{rel}(S)}\big(\mathrm{SD}_S\wedge D_S(f_2)\big)$ are equivalent, $\mathit{Sig}(f_1\cup f') = \mathit{Sig}(f_2 \cup f')$ for every $f' \subset S'$.

From this result, it is possible to take decisions at the local level. If two local fault modes f_1 and f_2 have the same local projected model, then:

- if $f_1 \subset f_2$, then for any $f' \subset \mathrm{COMPS} \setminus S$, we have $\mathit{MinSig}(f_2 \cup f') = \emptyset$, since $f_1 \cup f'$ has the same signature, and is included in $f_2 \cup f'$.
- if $f_1 \supset f_2$ the same result holds for f_1.
- if f_1 and f_2 are not related by inclusion, then for every $f' \subset \mathrm{COMPS} \setminus S$, we have $\mathit{MinSig}(f_1 \cup f') = \mathit{MinSig}(f_2 \cup f')$. The two local fault modes are aggregated and analyzed together.

In our experience, fault modes involving many components are not diagnosable. Detection of non-diagnosable fault modes at the local level can be of interest for identifying poorly diagnosable subsystems, or for checking the suitability of a component hierarchy for hierarchical diagnosis.

4.3 Example

Revisiting our example, let us assume that the components are combined in the following order: first A_1 and M_1 are aggregated, then M_2 is added. A_2 and M_3 are combined together, then combined with the rest of the components. Diagnosability analysis is performed incrementally, and the diagnosable fault modes that are not yet known to be (non-)discriminable are checked for discriminability.

Diagnosability The first subsystem to be analyzed is $\{A_1, M_1\}$. The possible local fault modes for this subsystem are $\emptyset$, $\{A_1\}$, $\{M_1\}$ and $\{A_1, M_1\}$. The relevant variables of $\{A_1, M_1\}$ are $\{a, c, y, f\}$. We have:

f	$\mathcal{P}_{\{a,c,y,f\}}(\mathrm{SD}_{\{A_1,M_1\}} \wedge D_{\{A_1,M_1\}}(f))$
$\emptyset$	$(f = a \cdot c + y)$
$\{A_1\}$	$\top$
$\{M_1\}$	$\top$
$\{A_1, M_1\}$	$\top$

(10)

Here $\top$ is the true constraint, meaning that variables a, c, y and f range freely over their respective domains. At this stage, it is already possible to discard $\{A_1, M_1\}$, since no fault mode containing these two components can ever be a minimal diagnosis for this system. $\{A_1\}$, $\{M_1\}$ are aggregated to be further analyzed in tandem. They are jointly referred to by the symbol $[A_1, M_1]$.

Then M_2 is combined with A_1 and M_1. We note $S = \{A_1, M_1, M_2\}$, the relevant variables for S are $\{a, b, c, d, y, f\}$ and we have:

f	$\mathcal{P}_{\mathit{rel}(S)}(\mathrm{SD}_S \wedge D_S(f))$
$\emptyset$	$(y = b \cdot d) \wedge (f = a \cdot c + y)$
$\{M_2\}$	$(f = a \cdot c + y)$
$\{[A_1, M_1]\}$	$(y = b \cdot d)$
$\{[A_1, M_1], M_2\}$	$\top$

(11)

At this stage, nothing can be discarded or aggregated. The analysis of A_2 and M_3 leads to the following result, in which $\{A_2\}$ and $\{M_3\}$ have been aggregated, and $\{A_2, M_3\}$ has been discarded. Relevant variables are $\{c, e, y, g\}$.

f	$\mathcal{P}_{\{c,e,y,g\}}(\mathrm{SD}_{\{A_2,M_3\}} \wedge D_{\{A_2,M_3\}}(f))$
$\emptyset$	$(g = c \cdot e + y)$
$\{[A_2, M_3]\}$	$\top$

(12)

Finally, the results in tables (11) and (12) are combined line by line, and projected on the set of relevant variables:

$$\begin{aligned} \mathit{rel}(\{A_1, A_2, M_1, M_2, M_3\}) &= \mathit{rel}(\mathrm{COMPS}) \\ &= V_{\mathrm{OBS}} = \{a, b, c, d, e, f, g\} \end{aligned}$$

f	$\mathcal{P}_{\mathrm{OBS}}(\mathrm{SD} \wedge D(f))$
$\emptyset$	$(f = a \cdot c + b \cdot d) \wedge (g = b \cdot d + c \cdot e)$
$\{M_2\}$	$(f - a \cdot c = g - c \cdot e)$
$\{[A_1, M_1]\}$	$(g = b \cdot d + c \cdot e)$
$\{[A_1, M_1], M_2\}$	$\top$
$\{[A_2, M_3]\}$	$(f = a \cdot c + b \cdot d)$
$\{[A_2, M_3], M_2\}$	$\top$
$\{[A_1, M_1], [A_2, M_3], M_2\}$	$\top$

(13)

In table (13), the first three lines are obtained from the combination of table (12)'s first line with table (11), and the three last rows from table (12)'s second line. It is possible to discard from this table all the fault modes represented by the last line, i.e., those that contain M_2 and one of A_1 or M_1 and one of A_2 or M_3. It is also possible to aggregate lines 3 and 5 and obtain the table:

f	$\mathcal{P}_{\mathrm{OBS}}(\mathrm{SD} \wedge D(f))$
$\emptyset$	$(f = a \cdot c + b \cdot d) \wedge (g = b \cdot d + c \cdot e)$
$\{M_2\}$	$(f - a \cdot c = g - c \cdot e)$
$\{[A_1, M_1]\}$	$(g = b \cdot d + c \cdot e)$
$\{[A_2, M_3]\}$	$(f = a \cdot c + b \cdot d)$
$\{[A_1, A_2, M_1, M_3], M_2\}$	$\top$

(14)

Table (14) reproduces the signature lattice illustrated in figure 2, except that non-diagnosable fault modes have been discarded. A fault mode is diagnosable if and only if it is mentioned in table (14).

Discriminability Table (14) provides some results about discriminability: fault modes represented by the same line are not discriminable. However, more information can be extracted from the system analysis. We use equation (6) to compute the constraints $\mathit{MinC}(f)$, the solutions of which form $\mathit{MinSig}(f)$.

f	$\mathit{MinC}(f)$
$\emptyset$	$(f = a \cdot c + b \cdot d) \wedge (g = b \cdot d + c \cdot e)$
$\{M_2\}$	$(f - a \cdot c = g - c \cdot e \neq b \cdot d)$
$\{[A_1, M_1]\}$	$(g = b \cdot d + c \cdot e) \wedge (f \neq a \cdot c + b \cdot d)$
$\{[A_2, M_3]\}$	$(f = a \cdot c + b \cdot d) \wedge (g \neq b \cdot d + c \cdot e)$
$\{[A_1, A_2, M_1, M_3], M_2\}$	$\mathit{AllDifferent}(f - a \cdot c, g - c \cdot e, b \cdot d)$

In this example, pairwise comparison of the constraints shows that none of the fault modes listed in the table is weakly discriminable from another fault mode. Two diagnosable fault modes are then either non-discriminable or discriminable.

Performance The algorithm has been designed under the hypothesis that constraints can be manipulated efficiently. We believe that the decision diagrams described in Sasao and Fujita [1996], in particular arithmetic decision diagrams, offer efficient tools for implementing this algorithm.

5. RELATED WORK

To our knowledge, diagnosability definition has always relied on faulty behavior specifications, whether direct or indirect as explained in the introduction. Work addressing the diagnosis capabilities of systems without fault specifications exist only for theoretic comparison of diagnosis approaches, or for sensor selection. However, all known definitions of diagnosability assume the presence of fault models, and are irrelevant as they would indicate the lowest possible degree of diagnosability for every system in our framework.

The closest work to our approach is described in Cordier et al. [2004], in which diagnosis algorithms from DX and FDI communities are compared. Correspondences are established between *Analytical Redundancy Relations (ARRs)* and potential conflicts. The study of potential conflicts provides almost directly information about diagnosability, in particular when restricted to minimal conflicts. An efficient and complete algorithm for computing minimal diagnoses by combining the model constraints is described. In our approach, we entrust the constraint computation to tools designed by the constraint solving community. These tools are more generic and may be less efficient; however, they provide additional constraint types that may ease the modeling.

The concepts of non-discriminability, weak discriminability and strong discriminability have been introduced by Travé-Massuyès et al. [2006]. This paper also introduces the idea that the normal mode is one of the many fault modes, and that detectability can be defined as discriminability from the normal mode. This paper reuses the hybrid framework of components and ARRs introduced by Cordier et al. [2004], with the same ability to deal with unspecified faulty behavior. However, these definitions are stated with respect to the set of all the possible observations under a fault mode f, which corresponds to our normal signature. Hence, a direct application of the definitions of Travé-Massuyès et al. [2006] would give the lowest possible level of diagnosability for every system, i.e., no pair of fault mode is discriminable, since all signatures overlap. Our definitions allow to distinguish between a system with poor observation features from a system with good ones, and offers better support for the optimization of a system for diagnosis at design time.

The incremental approach for checking diagnosability presented here is inspired by Pencolé and Cordier [2005] in which an incremental algorithm for checking diagnosability is described. The discrete event models of subsystems are aggregated one by one, and information not relevant to diagnosability is discarded after each aggregation. The algorithm precisely identifies the situations in which diagnosis is unable to discriminate faults. Yet, this approach requires the specification of the faulty behavior as part of the component model, as do all event-based diagnosis approaches we are currently aware of.

Dressler and Struss [2003] and Pucel et al. [2007] provide diagnosability analysis approaches for models using constraints over discrete variables. Both rely on a specification of faulty behavior. Dressler and Struss [2003] introduce the properties of necessary and possible discriminability, the purpose being to identify operating commands that allow to discriminate faults. Pucel et al. [2007] describe a hierarchical approach to diagnosability, and a hierarchical algorithm for checking it. The concept of partial fault mode used in that approach is similar to the local fault modes presented in this paper, as both describe the fault mode of a subsystem.

6. CONCLUSION

This paper addresses the problem of diagnosability analysis using constraint based models, with unrestricted faulty behavior. Diagnosis relies on the decomposition of the system into components to infer from symptoms which parts of the system comply to their normal behavior and which do not. Diagnosability is defined with respect to this diagnostic reasoning, and predicts which sets of components can be minimal diagnoses for some possible observation.

When a fault mode is not diagnosable, this means that when the system is in this fault mode, a diagnosis algorithm will generally suspect smaller but incomplete sets of faults. It is up to the system designer to accept a given degree of diagnosability, or to modify the system or the model in order to increase diagnosability. Our approach also identifies minimal diagnoses that are non-discriminable. Such information can be a useful input for model abstraction, which has been linked to potential conflicts by Perrot and Travé-Massuyès [2007].

Our diagnosability approach is unique since it assumes that diagnosis may output an incorrect explanation when the system is in a non-diagnosable fault mode. This impacts strongly on the decisions that follow diagnosis, and self-healability as defined in Cordier et al. [2007] needs to be adapted to account for such a diagnosis approach.

In the domain of software programming, model-based diagnosis can be performed on an abstract representation of a software program as described by Mayer and Stumptner [2008]. A comfortable range of abstraction techniques are available, offering various degrees of precision and computational cost. Faulty behavior is not available in such context, and diagnosability analysis can be used to help choosing a suitable abstraction among the possible ones. In this context, our work should be seen as the first step of an approach to assess diagnosability of programs with respect to a library of models in order to select a suitable abstraction for automated debugging, tailored to the program under consideration. This is a significant problem that has so far not been addressed adequately.

REFERENCES

Marie-Odile Cordier, Philippe Dague, François Lévy, Marcel Staroswiecki, and Louise Travé-Massuyès. Conflicts versus analytical redundancy relations: a comparative analysis of the model based diagnosis approach from the artificial intelligence and automatic control perspectives. *IEEE Transactions on Systems, Man, and Cybernetics Part B*, 34(5):2163–2177, October 2004.

Marie-Odile Cordier, Louise Travé-Massuyès, and Xavier Pucel. Comparing diagnosability in continuous and discrete-event systems. In *Proceedings of the 17th International Workshop on Principles of Diagnosis, DX'06*, pages 55–60, 2006.

Marie-Odile Cordier, Yannick Pencolé, Louise Travé-Massuyès, and Thierry Vidal. Self-healability = diagnosability + repairability. In *Proceedings of the 18th International Workshop on Principles of Diagnosis, DX'07*, pages 265–272, Nashville, TN, USA, May 2007.

Oskar Dressler and Peter Struss. A toolbox integrating model-based diagnosability analysis and automated generation of diagnostics. In *Proceedings of the 14th International Workshop on Principles of Diagnosis, DX'03*, 2003.

Walter Hamscher, Luca Console, and Johan de Kleer. *Readings in model-based diagnosis.* Morgan Kaufmann Publishers Inc, 1992.

Wolfgang Mayer and Markus Stumptner. Evaluating models for model-based debugging. In *23rd IEEE/ACM International Conference on Automated Software Engineering (ASE 2008)*, pages 128–137, L'Aquila, Italy, September 2008. IEEE Computer Society Press. ISBN 978-1-4244-2187-9.

Yannick Pencolé and Marie-Odile Cordier. A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks. *Artificial Intelligence Journal*, 164 (1–2):121–170, 2005.

Fabien Perrot and Louise Travé-Massuyès. Choosing abstractions for hierarchical diagnosis. In Gautam Biswas, Xenofon Koutsoukos, and Sherif Abdelwahed, editors, *Proceedings of the 18th International Workshop on Principles of Diagnosis, DX'07*, pages 354–359, Nashville, TN, USA, May 2007.

Xavier Pucel. *Another Point of View on Diagnosability.* PhD thesis, Université de Toulouse ; Institut National des Sciences Appliquées (INSA), CNRS ; LAAS ; Université de Toulouse ; 7, avenue du Colonel Roche, F-31077 Toulouse, France, December 2008.

Xavier Pucel, Stefano Bocconi, Claudia Picardi, Daniele Theseider Dupré, and Louise Travé-Massuyès. Diagnosability analysis for web services with constraint-based models. In Gautam Biswas, Xenofon Koutsoukos, and Sherif Abdelwahed, editors, *Proceedings of the 18th International Workshop on Principles of Diagnosis, DX'07*, pages 360–367, Nashville, TN, USA, May 2007.

Meera Sampath, Raja Sengupta, Stephane Lafortune, Kasim Sinnamohideen, and Demosthenis Teneketzis. Diagnosability of discrete event system. *IEEE Transactions on Automatic Control*, 40(9):1555–1575, September 1995.

Tsutomu Sasao and Masahiro Fujita, editors. *Representations of Discrete Functions.* Kluwer Academic Publishers, Norwell, MA, USA, 1996. ISBN 0792397207.

Louise Travé-Massuyès, Teresa Escobet, and Xavier Olive. Diagnosability analysis based on component-supported analytical redundancy relations. *IEEE Transactions on Systems, Man, and Cybernetics Part A*, 36(6):1146–1160, November 2006.