FUNDAMENTALS OF MODEL-BASED DIAGNOSIS

Johan de Kleer, James Kurien

PARC
3333 Coyote Hill Road
Palo Alto, CA, 94304
{*dekleer, jkurien*}*@parc.com*

Abstract: Over the last 25 years, the Computer Science community and particularly the Artificial Intelligence community, have developed a framework for system diagnosis, called Model-Based Diagnosis. This framework is extremely general and covers a broad range of capabilities including detecting malfunctions, isolating faulty components, handling multiple faults, identifying repair actions, and automatically generating embedded software. This field grew independently of the fault detection and isolation community (FDI) and has developed its own terminologies and conventions. This paper is an attempt to present the fundamental concepts of Model-Based Diagnosis (MBD) in one place and in one consistent terminology, and thus make the field much more accessible to the FDI community.

Keywords: Model-based diagnosis, consistency-based diagnosis, constraint-propagation, qualitative models, probabilistic inference, analytical redundancy relations.

1. INTRODUCTION

Like FDI, MBD seeks to develop algorithms which can perform diagnostic tasks on complex systems without human intervention. MBD techniques have been used to automatically diagnose and mitigate failures on-board spacecraft (Williams and Nayak, 1996; Bernard *et al.*, 1998), diagnose problems in automotive systems (Sachenbacher *et al.*, 2000) and determine optimal placement of sensors during design (Mauss *et al.*, 2000). They have been made small enough to be deployed on embedded systems, scalable enough to be applied to systems with tens of thousands of components, and robust enough to be offered as commercial products. The distinguishing features of the MBD approach are an emphasis on general diagnostic reasoning engines that perform a variety of diagnostic tasks via on-line reasoning, and inference of a system's global behavior from the automatic combination of local models of its components. This is illustrated in Figure 1. For example, a MBD engine can be provided a schematic of a circuit, values of some of its inputs and outputs and it can determine from only that information whether the circuit is malfunctioning, which components might be faulty and what additional information need to be gathered (if any) to identify the faulty components with relative certainty.

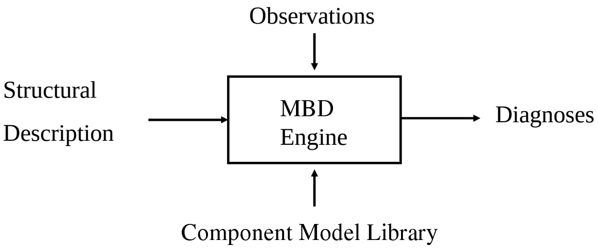

Fig. 1. Basic Perspective of MBD.

The component library used by an MBD engine describes the laws which govern the behavior of the components. A resistor obeys Ohm's law, a multiplier component obeys the constraint that its output is the product of its inputs. Once provided a component model library, the MBD engine should be able to diagnose any system constructed out of known components. A very important modeling principle is that the component models obey a *no-function-in-structure-principle* — a component model cannot make assumptions about the kinds of systems it may appear within or the kinds of inputs it is normally provided and instead must describe the behavior of the component in all possible systems with all possible inputs. Hence, MBD models are *compositional* — the model of a combination of two systems is directly constructed from the models of the constituent systems.

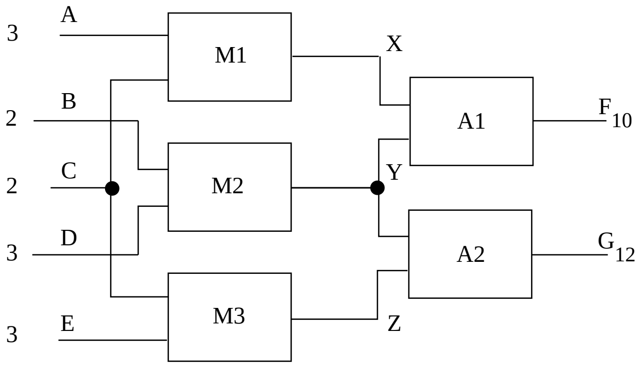

Fig. 2. A, B, C, D and E are input terminals, F and G are output terminals, X, Y and Z are internal probe points, M_1, M_2 and M_3 are multipliers and A_1 and A_2 are adders.

Consider the system illustrated in Figure 2. (Although MBD can be applied to a wide variety of very large continuous and discrete systems, this paper will employ a few very simple systems to exposit the basic ideas.) To model this system, the model library would include:

$$(MULTIPLIER\ m\ i\ j\ k) \rightarrow k = i \times j$$

$$(ADDER\ a\ i\ j\ k) \rightarrow k = i + j$$

An MBD engine would be provided a structural description which might simply be characterized as:

```
(MULTIPLIER M1 A C X)
(MULTIPLIER M2 B D Y)
(MULTIPLIER M3 C E Z)
(ADDER A1 X Y F)
(ADDER A2 Y Z G)
```

The inputs to the diagnostic engine would be observations from the system:

```
A=3, B=2, C=2, D=3, E=3, F=10, G=12
```

From this information the MBD engine would determine the following sets of components could be faulted: $\{A_1\}, \{M_1\}, \{A_2, M_2\}, \{M_2, M_3\}$ and any of their supersets. In addition, the most informative place to measure next is X because it distinguishes between two single faults.

This example illustrates some of the basic properties of MBD:

- A system model is provided in terms of components and their interconnections.
- The component models describe how each component behaves, not how to diagnose them.
- A domain-independent reasoning engine calculates the diagnoses from the model.
- The system may have multiple dependent or independent faults.
- MBD can propose additional measurements to differentiate among diagnoses.
- MBD does not do any precomputation - it is entirely online.

Many of the seminal papers of Model-Based Diagnosis can be found in (Hamscher *et al.*, 1992). This introductory paper draws from the first author's previous work (de Kleer and Williams, 1987),(de Kleer and Williams, 1989),(de Kleer *et al.*, 1992) and (Forbus and de Kleer, 1992). The first three articles can be found in (Hamscher *et al.*, 1992).

2. FORMAL DEFINITIONS

Unlike FDI, MBD analyses use logical inference. Before defining MBD more formally, we illustrate how an MBD engine draws inferences. Consider the example system. Given inputs $A = 3$, $B = 2$, $C = 2$, $D = 3$, and $E = 3$, by simple calculation (i.e., the inference procedure), $X = 6$, $Y = 6$, and $F = X + Y = (A \times C) + (B \times D) = 12$. Intuitively, a *symptom* is any difference between a prediction made by the inference procedure and an observation. Since, F is measured to be 10, "F is observed to be 10, not 12" is a symptom. In MBD, the predictions need not be causal. For example, in the case where the value of input B were unknown, MBD would predict $Y = 4$ as that is the only value Y could be if one of the inputs to adder A_1 were 4 and its output 10.

The symptoms drive diagnostic reasoning. Each symptom indicates one or more components may be faulted. Intuitively, a *conflict* is a set of components which underly a symptom. In the example, a conflict is a set of components which cannot all be functioning correctly. Consider the symptom "F is observed to be 10, not 12." The prediction that $F = 12$ depends on the correct operation of A_1, M_1, and M_2, *i.e.*, if A_1, M_1, and M_2 were correctly functioning, then $F = 12$. Since F is not 12, at least one of A_1, M_1, and M_2 is faulted. Thus the set $\{A_1, M_1, M_2\}$ is a conflict for the symptom. The set $\{A_1, A_2, M_1, M_2\}$, and

Diagnosis	Sentence
$\mathcal{D}(\{A_1\},\{A_2,M_1,M_2,M_3\}):$	$AB(A_1)\wedge\neg AB(A_2)\wedge\neg AB(M_1)\wedge\neg AB(M_2)\wedge\neg AB(M_3)$
$\mathcal{D}(\{M_1\},\{A_1,A_2,M_2,M_3\}):$	$AB(M_1)\wedge\neg AB(A_1)\wedge\neg AB(A_2)\wedge\neg AB(M_2)\wedge\neg AB(M_3)$
$\mathcal{D}(\{M_2,M_3\},\{A_1,A_2,M_1\}):$	$AB(M_2)\wedge AB(M_3)\wedge\neg AB(A_1)\wedge\neg AB(A_2)\wedge\neg AB(M_1)$
$\mathcal{D}(\{A_2,M_2\},\{A_1,M_1,M_3\}):$	$AB(A_2)\wedge AB(M_2)\wedge\neg AB(A_1)\wedge\neg AB(M_1)\wedge\neg AB(M_3).$

Fig. 3. Four minimal diagnoses with the corresponding logical sentences.

any other superset of $\{A_1,M_1,M_2\}$ are conflicts as well; however, no subsets of $\{A_1,M_1,M_2\}$ are necessarily conflicts since all the components in the conflict were needed to predict the value at F.

A *diagnosis* is a particular hypothesis for how the system differs from its model. For example "A_2 and M_2 are broken" is a diagnosis which explains the two symptoms observed for the example system. The size of the initial diagnosis space is exponential in the number of system components. Any component could be working or faulty, thus the diagnosis space for the system initially consists of $2^5 = 32$ diagnoses. Ultimately, the goal of diagnosis is to identify, and refine, the set of diagnoses consistent with the observations thus far. Recall that each conflict is a set of components such that at least one must be faulty. Thus each set representing a diagnosis must have a non-empty intersection with every conflict. This observation is key to the algorithms MBD has developed for efficiently finding diagnoses.

These diagnostic concepts can be defined more formally using First-Order Logic within the framework of (Reiter, 1987) and (de Kleer *et al.*, 1992).

Definition 1. A system is a triple (SD,COMPS, OBS) where:

(1) SD, the system description, is a set of first-order sentences.
(2) COMPS, the system components, is a finite set of constants.
(3) OBS, a set of observations, is a set of first-order sentences.

The definition of diagnosis is built up from the notion of abnormal. Intuitively, the model for the correct behavior of a component is written as "If the resistor is not abnormal, then its behavior is modeled by Ohm's Law." The standard MBD convention is that $AB(c)$ is a literal which holds when component $c\in$COMPS is abnormal. The adder and multiplier of Figure 2 can be modeled as followed.

$$ADDER(x)\rightarrow$$
$$[\neg AB(x)\rightarrow out(x) = in1(x)+in2(x)]$$
$$MULTIPLIER(x)\rightarrow$$
$$[\neg AB(x)\rightarrow out(x) = in1(x)\times\ in2(x)]$$

Assume that SD is extended with the appropriate axioms for arithmetic, etc. The logical sentence describing adders can be read as: If x is an adder, then if x is not abnormal, then the output of the adder is the sum of its inputs. No model of how the adder behaves when it is abnormal has been introduced. Therefore, in this model formulation it is impossible to ever prove that an adder is unfaulted.

A diagnosis specifies whether each component of a system is abnormal or not. More formally, a diagnosis is defined as follows:

Definition 2. Given two sets of components Cp and Cn define $\mathcal{D}(Cp,Cn)$ to be the conjunction:

$$\Big[\bigwedge_{c\in Cp} AB(c)\Big]\wedge\Big[\bigwedge_{c\in Cn}\neg AB(c)\Big].$$

A diagnosis is a sentence describing one possible state of the system, where this state is an assignment of the status normal or abnormal to each system component. A consistency-based diagnosis is a diagnosis whose assignments of each component c to $AB(c)$ or $\neg AB(c)$ are consistent with the system model and current observations. More formally, a consistency-based diagnosis is defined as follows (the formal way of saying "A is consistent with B" is to say $A\cup B$ is satisfiable).

Definition 3. Let $\Delta\ \subseteq$COMPS. A diagnosis for (SD,COMPS,OBS) is $\mathcal{D}(\Delta, COMPS-\Delta)$ such that the following is satisfiable:

$$SD\cup OBS\cup\{\mathcal{D}(\Delta, COMPS-\Delta)\}$$

There may be an exponential number of diagnoses ($2^{|COMPS|}$). In the example there are four minimal diagnoses as illustrated in Figure 3. Formally, a minimal diagnosis is defined as follows:

Definition 4. A diagnosis $\mathcal{D}(\Delta, COMPS-\Delta)$ is a minimal diagnosis iff for no proper subset Δ' of Δ is $\mathcal{D}(\Delta', COMPS-\Delta')$ a diagnosis.

Conflicts provide an intermediate step in determining the diagnoses. Intuitively, to identify the diagnoses for a system, MBD first determines the symptoms, then determines the minimal conflicts, and finally determines the minimal diagnoses directly from the minimal conflicts. Recall that a conflict is a set of components which cannot all be operating properly. That is to say, at least

	Inference	Source
1	$\neg AB(M_1) \rightarrow out(M_1) = in1(M_1) \times\ in2(M_1)$	Library
2	$\neg AB(M_1) \rightarrow out(M_1) = 6$	$in1(M_1) = 3$; $in2(M_1) = 2$
3	$\neg AB(M_2) \rightarrow out(M_2) = in1(M_2) \times\ in2(M_2)$	Library
4	$\neg(M_2) \rightarrow out(M_2) = 6$	$in1(M_2) = 2$; $in2(M_2) = 3$
5	$\neg AB(A_1) \rightarrow out(A_1) = in1(A_1) \times\ in2(A_1)$	Library
6	$in1(A_1) = out(M_1)$ and $in2(A_1) = out(M_2)$	SD
7	$\neg AB(M_1) \wedge \neg AB(M_2) \wedge \neg AB(A_1) \rightarrow out(A_1) = 12$	2,4,6
8	$\neg AB(M_1) \wedge \neg AB(M_2) \wedge \neg AB(A_1) \rightarrow \bot$.	$out(A_1) = 10$
9	$AB(M_1) \vee AB(M_2) \vee AB(A_1)$	$(x \rightarrow \bot) \equiv \neg x$ and $\neg(\neg x \wedge \neg y) \equiv (x \vee y)$

Fig. 4. Inference of the conflict $AB(M_1) \vee AB(M_2) \vee AB(A_1)$, where $\bot$ is the symbol for false.

one of the components must be abnormal. Thus the conflict $\{A_1, M_1, M_2\}$ can also be represented $AB(A_1) \vee AB(M_1) \vee AB(A_2)$. The next three definitions introduce conflicts more formally.

Definition 5. An AB-literal is $AB(c)$ or $\neg AB(c)$ for some c $\in$ COMPS.

Definition 6. An AB-clause is a disjunction of AB-literals containing no complementary pair of AB-literals.

Definition 7. A conflict of (SD,COMPS,OBS) is an AB-clause entailed by SD $\cup$ OBS.

There are typically an exponential number of possible conflicts, thus it is important to characterize the space of possible conflicts by minimal conflicts.

Definition 8. A minimal conflict of (SD,COMPS, OBS) is a conflict no proper sub-clause of which is a conflict of (SD,COMPS,OBS).

Figure 4 illustrates the derivation of the conflict $AB(A_1) \vee AB(M_1) \vee AB(M_2)$ from the example of Figure 2. Another conflict stemming from the observation $F = 10$ is $AB(A_1) \vee AB(A_2) \vee AB(M_1) \vee AB(M_3)$. Coincidentally, these are the only two minimal conflicts for this example. The minimal diagnoses for a system can be determined from the minimal conflicts alone. No other information about the system is needed:

Lemma 1. Suppose that Π is the set of minimal conflicts of (SD,COMPS,OBS), and that Δ is a minimal set such that,

$$\Pi \cup \{ \bigwedge_{c \in COMPS-\Delta} \neg AB(c)\}$$

is satisfiable. Then $\mathcal{D}(\Delta, COMPS - \Delta)$ is a minimal diagnosis.

This result justifies the intuitive scheme presented earlier in this section. Figure 5 illustrates the space of possible diagnoses resulting from these two conflicts. For brevity, diagnoses are written using $[\ldots]$ notation so $[A_2, M_2]$ represents $\mathcal{D}(\{A_2, M_2\}, \{A_1, M_1, M_3\})$ and conflicts are written using $\langle \ldots \rangle$ notation so $\langle A_1, A_2, M_1, M_3 \rangle$ represents $AB(A_1) \vee AB(A_2) \vee AB(M_1) \vee AB(M_3)$. Specific algorithms for generating diagnoses are presented in a subsequent section. Intuitively, starting at the empty diagnosis [] and moving upward along the arcs, any node whose label names at least one component in each conflict is a diagnosis. Thus the four circled nodes are the minimal diagnoses.

3. RELATIONSHIP TO ANALYTICAL REDUNDANCY

As shown in (Cordier *et al.*, 2000) there is a direct link between MBD conflicts and FDI analytical redundancy relations. An analytical redundancy relation (ARR) expresses a constraint among possible observations that holds when the system is working correctly. The FDI structure is the minimal set of components whose models have to be satisfied for the ARR to be satisfied. Some of the ARRs for Figure 2 and their structures are:

	ARR	Structure
1	$f - ac - bd = 0$	A_1,M_1,M_2
2	$g - ce + bd = 0$	A_2, M_2,M_3
3	$f - g - ca + ce = 0$	A_1,A_2,M_1,M_3
4	$f - g + ce - x = 0$	A_1,A_2,M_3
5	$y - bc = 0$	M_2

If some ARR relation does not hold, then the system is faulted and the degree to which the relations are violated are the residuals. Table 1 describes part of the signature matrix for the system of Figure 2. Note that A_1 appears in the structure for ARR's 1, 3 and 4. Thus in the signature table, a one indicates that the pattern for indicating a failure of A_1 includes a residual at the corresponding ARR. Under the FDI single fault assumption if these ARR's all have residuals, A_1 is a diagnosis.

ARR's can be defined in the MBD logical framework, although there is not enough space to

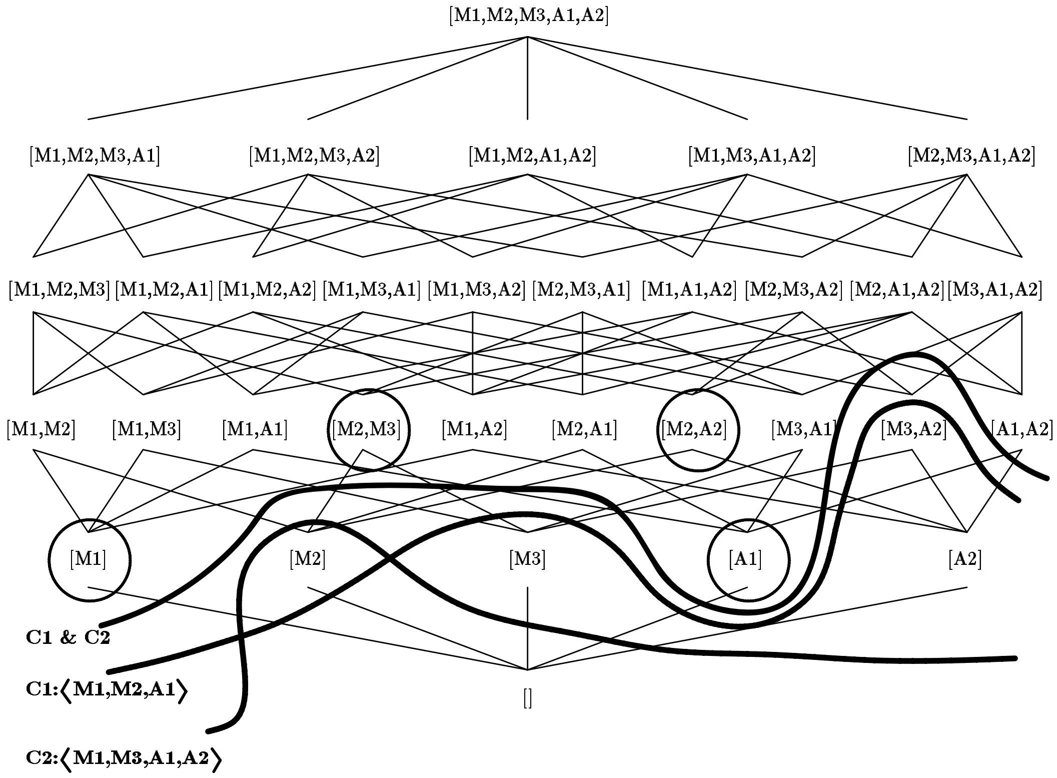

Fig. 5. This describes the space of possible diagnoses after identifying the two conflicts for Figure 2. For brevity, diagnoses are labeled by their faulted components alone. The node at the top is the diagnosis in which every component is faulted. The node at the bottom is the eliminated diagnoses where no component is faulted. The circled nodes are the minimal diagnoses, and every node which can be traced upwards along any graph edge is also a diagnosis. Each conflict eliminates from consideration any diagnosis which does not contain one of its faulted components, so any diagnosis below a conflict curve is eliminated.

Table 1. Partial signature matrix.

ARR	$[A_1]$	$[A_2]$	$[M_1]$	$[M_2]$	$[M_3]$	$[M_1, M_2]$
1	1	0	1	1	0	1
2	0	1	0	1	1	1
3	1	1	1	0	1	1
4	1	1	0	0	1	0
5	0	0	0	1	0	1

present all the formalities. Intuitively, an ARR is an equation that can be inferred from the system description under the assumption that no components are failed.

Definition 9. An ARR of a system is a mathematical equation $f(x_1, \ldots, x_n) = 0$ where x_i are observable system variables that is entailed by SD and [], where no such equation on a proper subset of $\{x_1, \ldots, x_n\}$ is entailed.

The ARR's with their structures are illustrated in Figure 6. In the MBD framework, the ARR structure can be defined as:

Definition 10. The structure of an ARR is a set $S \subset COMPS$ such that $\bigwedge_{c \in S} \neg AB(c) \rightarrow f(x_1, ..., x_n) = 0$, is entailed by SD and no subset of S has this property.

The signature matrix for a system can be constructed directly from the ARRs. Given diagnosis $\mathcal{D}(Cp, Cn)$ and ARR $\bigwedge_{c \in S} \neg AB(c) \rightarrow f(x_1, ..., x_n) = 0$, the signature matrix can be constructed as follows:

$$(ARR, \mathcal{D}(Cp, Cn)) \rightarrow 1 \quad \text{iff} \quad S \cap Cp \neq \emptyset.$$

The structure of each violated ARR (having a non-zero residual) corresponds directly to a conflict. So FDI and MBD are analogous in that both derive the same intermediate information from which the diagnoses are subsequently determined. The biggest difference is that MBD never computes a signature matrix either offline or online. The reason behind this is that MBD does not assume any limit on the number of faults in the system, and it must avoid constructing an exponentially sized data structure. Instead, MBD algorithms (see Section 6) directly compute the relevant diagnoses for a system without enumerating the $2^{|COMPS|}$ possibilities. Thus, MBD has an advantage over FDI in that it can consider

1	$\neg AB(M_1) \wedge \neg AB(M_2) \wedge \neg AB(A_1)$	$\rightarrow$	$f - ac - bd = 0$
2	$\neg AB(M_2) \wedge \neg AB(M_3) \wedge \neg AB(A_2)$	$\rightarrow$	$g - ce + bd = 0$
3	$\neg AB(M_1) \wedge \neg AB(A_1) \neg AB(A_2) \wedge \neg AB(M_3)$	$\rightarrow$	$f - g - ca + ce = 0$
4	$\neg AB(A_1) \wedge \neg AB(A_2) \wedge \neg AB(M_3)$	$\rightarrow$	$f - g + ce - x = 0$
5	$\neg AB(M_2)$	$\rightarrow$	$y - bc = 0$

Fig. 6. ARR's in Logical Form

multiple faults without necessarily paying the exponential price.

MBD does not employ the notion of exoneration used in FDI approaches which assume that if an ARR is satisfied (residual is 0), then the components in the structure are exonerated. In MBD's logical formulation, this assumption would be equivalent to writing ARR sentences as:

$$\bigwedge_{c \in S} \neg AB(c) \equiv f(x_1, ..., x_n) = 0.$$

Unlike the implication form (using $\rightarrow$ instead of $\equiv$) of ARRs, these sentences do not logically follow from SD and adding these typically render $SD \cup OBS$ unsatisfiable, effectively making the null set a conflict. Assume that in our example, M_2 is faulted with behavior $Y = B \times C - 2$ and A_2 is faulted with behavior $G = Y \times Z + 2$. FDI concludes that M_2, M_3 and A_2 are unfaulted because the ARR 2 is satisfied, and the diagnosis $[M_2, A_2]$ is incorrectly eliminated. Thus, if a measurement for Y is included in the observations, a logical inconsistency will occur (*i.e.*, M_2 is both faulted and unfaulted). To summarize, FDI approaches assume that multiple faults cannot compensate for each other, while MBD do not.

A possible criticism of MBD is that it does not eliminate extremely unlikely diagnoses and component faults, and thus treats plausible and implausible diagnoses equally. The next section describes how MBD incorporates a probabilistic overlay on the logical framework such that unlikely diagnoses are retained, but receive very low posterior probability. We return to the specific example of compensatory faults after introducing probabilities. With probabilities incorporated, MBD will never incorrectly eliminate a possible diagnosis and will assign components occurring in compensatory diagnoses or with exonerating behavior very low posterior fault probabilities.

4. INTRODUCING PROBABILITIES

MBD makes no presupposition about the number of faults in a system. As a consequence, the logical framework provides no way to limit the number of diagnoses. Even the number of minimal diagnoses for a system can grow exponentially. Instead, MBD uses probabilities to rank diagnoses and limit the number of diagnoses that need to be identified. An MBD engine can provide a sequence of possible diagnoses ordered by likelihood. This probability can be used for many purposes, such as deciding the probability of one diagnosis is sufficiently high to forgo further computation or evidence gathering.

Without any observations, the probability of a particular diagnosis correctly characterizing the actual state of the system is computed from the prior probabilities of component failure (which can be obtained from their manufacturer, by observation, or from previous experience with this type of system). Although MBD can be extended to handle dependent faults, we make the presupposition that components fail independently. The initial probability that a particular diagnosis $\mathcal{D}(Cp, Cn)$ is correct is thus:

$$p(\mathcal{D}) = \prod_{c \in C_p} p(c) \prod_{c \in C_n} (1 - p(c)),$$

where $p(c)$ is the prior probability that component c is faulted. The fundamental equation which determines the posterior probability of a diagnosis D after gathering evidence is Bayes Rule. In the case where evidence is a measurement of a particular value $x = v$, Bayes Rule takes the following form:

$$p(D|x = v) = \frac{p(x = v|D)p(D)}{p(x = v)}.$$

The term $p(D)$ can be computed from the prior probabilities of component failures. The term $p(x = v)$ is a normalizing term that is equivalent for all D and thus need not be computed directly. Thus the only term remaining to be evaluated in the equation is $p(x = v|D)$. Fortunately, this probability can be directly determined from the formal logical analysis:

$$p(x = v|D) = 1 \quad \text{if} \quad x = v \;\; \text{follows from} \;\; D, SD,$$

$$p(x = v|D) = 0 \quad \text{if} \quad D, SD, (x = v) \text{ are inconsistent.}$$

In the second case above, measuring $x = v$ rules out D as a diagnosis. In the case where neither holds we use the heuristic

$$p(x = v|D) = \frac{1}{m},$$

where m is the number of possible values x can take on (assuming x takes discrete values). This heuristic captures the assumption that without evidence, every possible value for a variable is equally likely. Although this assumption is clearly false because interconnected components co-constrain variables, it is a good average approximation. For subsequent observations, Bayes Rule is applied sequentially.

Consider the system of Figure 2. Suppose that M_2 is faulted with behavior $Y = B \times C - 2$ and A_2 is faulted with behavior $G = Y \times Z + 2$. This is an highly unlikely compensatory double fault. Assume that components fail with initial probability .01 and that $m = 16$. Before any output measurements are made the most probable diagnosis is [], $p([]) = .951$. Suppose F is measured with result $F = 10$. The most probable diagnoses are now $[A_1]$, $[M_1]$, and $[M_2]$, all having probability .323. Note the diagnoses $[A_2]$ and $[M_3]$ predict that $F = 12$ and therefore $p([A_2]|F = 12, ...) = 0$ and $p([M_3)|F = 12, ...) = 0$. None of the three leading diagnoses predict $F = 12$, $p(F = 12|[A_1], ...) = p(F = 12|[M_1], ...) = p(F = 12|[M_2], ...) = \frac{1}{16}$. In fact, none of the remaining possible diagnoses predicts $F = 12$. At this point the actual value of m is thus irrelevant as it is a constant factor for all diagnoses.

Next, suppose G is measured with result $G = 12$. The most probable diagnoses are now $[A_1]$ and $[M_1]$ both with probability 0.478. One of the reasons these two diagnoses receive such a high probability is that both of them predict $G = 12$.

Next, suppose X is measured with result $X = 6$. This results in a single high-probability diagnosis $[A_1]$ with probability .942. The next seven most likely diagnoses are: $[A_1, M_3]$, $[A_1, M_2]$, $[A_1, M_1]$, $[A_1, A_2]$, $[A_2, M_2]$,and $[M_2, M_3]$, all with probability 0.00951. Note that although $[A_1]$ is a high-probability single fault, the component A_1 is not faulted.

Next, suppose Y is measured with result $Y = 4$. M_2 is now necessarily faulted and at least one other fault exists.

Finally, suppose Z is measured with result $Z = 6$. There are six remaining diagnoses: $[A_2, M_2]$ with probability 0.970, $[A_2, M_1, M_2]$, $[A_1, A_2, M_2]$, $[A_2, M_2, M_3]$, with probabilities 0.0098, $[A_1, A_2, M_1, M_2]$, $[A_1, A_2, M_2, M_3]$, $[A_2, M_1, M_2, M_3]$, with probabilities 0.0001, and $[A_1, A_2, M_1, M_2, M_3]$ with probability 0.000001. Components M_2 and A_2 are necessarily faulted, and A_1, M_1, and M_3 are possibly faulted, each with probability .01. These three components cannot be completely exonerated because the simple models do not provide any indication how they behave when faulted.

5. PROBING STRATEGIES

The current set of observations may be insufficient to identify the faulty components with adequate confidence. To reduce the set of diagnoses, MBD collects additional observations which help isolate the actual fault. Collecting additional observations may be expensive. The initial set of observations may have been provided by direct sensors that were essentially free to obtain or already represents an incurred cost, while gathering additional observations may require a technician placing an instrument in the physical system itself. To simplify matters we assume every probe is of equal cost and thus the task is to identify the faulty components with a minimum number of probes, on average.

A practical approach utilizes Shannon entropy to estimate the cost of identifying the correct diagnosis. Given a set of diagnoses, S, the Shannon entropy is,

$$H = -\sum_{D \in S} p(D) log\ p(D),$$

where the $p(D)$ is the probability that diagnosis D is correct and the cost of each probe is identically 1. Recall from Section 4 that MBD can determine the probability of every possible measurement outcome as well as the diagnosis probability distribution resulting from such an outcome. Therefore, we can determine the expected entropy after measuring some system variable — which is the same as estimating the number of additional probes required after the measurement is made. Thus, we have enough information at hand to compute,

$$H_e(x_i) = \sum_{k=1}^{m} p(x_i = v_{ik}) H(x_i = v_{ik}),$$

where $p(x_i = v_{ik})$ is the probability of that x_i is measured to be v_{ik} and $H(x_i = v_{ik})$ is the entropy of the resulting distribution of diagnosis probabilities. It can be shown after considerable algebra that the change in entropy resulting from measuring x_i is,

$$\sum_{k=1}^{m} p(x_i = v_{ik}) log\ p(x_i = v_{ik}) + p(U_i) log\ m,$$

where m is the number of possible values for x_i and $p(U_i)$ is the sum of the probabilities of all the diagnoses that do not predict a value for x_i. Intuitively, this expression represents the information (negative entropy) of the distribution of outcomes of measuring x. For convenience, we add 1 to define $\$(x_i)$, the cost of measuring x_i. It is possible to extend this one-step lookahead approach to multiple steps, however the computational cost usually becomes excessive.

With this probing strategy, MBD can quickly diagnose the faults in our example. As in the previous section, assume all components fail with probability .01, that the faults are M_2 with behavior $Y = B \times C - 2$, and A_2 with behavior $G = Y \times Z + 2$, and that $m = 16$. If only the inputs are known, the lowest cost possible measurements are F and G. This framework will always tend to suggest measuring outputs first because those values have a broader distribution of possible discrepant values. Suppose $F = 10$, then the next best measurement is G. The subsequent measurements are $X = 6$ and $Y = 4$. The expected costs for the measurements are listed in Table 2. Note that for any particular fault the probing strategy may not yield the best result as the goal is to produce the best cost averaged over all possible faults.

Table 2. Expected costs during sequential diagnosis of Figure 2 (when the outputs are not initially provided).

	initial	F=10	G=12	X=6	Y=4
F	.88	1	1	1	1
G	.88	.28	1	1	1
X	.95	.34	.28	1	1
Y	.95	.34	.94	.90	1
Z	.95	95	.97	.94	.141

6. EFFICIENT ALGORITHMS

There is an extensive literature on developing efficient algorithms for MBD. The logical framework we have developed precisely define the notions of conflict and diagnosis, and thus provides a clear specifications for the algorithms. Nevertheless, great care must be taken to define efficient algorithms, otherwise practical MBD will be computationally infeasible.

6.1 Complete Algorithms

There are two key inferential processes in MBD: reasoning from the observations to determine the discrepancies and their associated conflicts and computing the diagnoses from those conflicts. Consider the case of computing the diagnoses from the conflicts in the common situation where there are no fault models. In this case, we draw on definitions and algorithms developed for logical circuit design (Tison, 1967)(Kohavi, 1978).

Definition 11. Suppose Σ is a set of propositional formulas. A satisfiable conjunction of literals π (i.e., a conjunction containing no pair of complementary literals) is an implicant of Σ iff π entails each formula in Σ. π is a prime implicant of Σ if no other implicant contains a subset of π's literals.

Prime implicates can be defined analogously as disjunctions.

Theorem 1. $\mathcal{D}(\Delta, COMPS - \Delta)$ is a minimal diagnosis of (SD,COMPS,OBS) iff $\bigwedge_{c \in \Delta} AB(c)$ is a prime implicant of the set of minimal conflicts of (SD,COMPS,OBS).

It is important to focus on minimal conflicts and minimal diagnoses, because the number of possible conflicts and diagnoses is always exponential in $|COMPS|$. Some of the best prime implicant/implicate algorithms are based on Boolean Decision Diagram algorithms (BDDs) (Simon and del Val, 2001).

Most MBD implementations are not able to draw all possible inferences from $SD \cup OBS$ (*i.e.*, they are incomplete) and therefore cannot discover all possible conflicts. Instead, they use sound but incomplete techniques such as described in the next two subsections.

6.2 Approximate Algorithms

A common MBD inference architecture uses a combination of constraint propagation and truth maintenance. Constraint propagation presumes components are modeled as constraints and repeatedly finds some constraint which could only be satisfied by forcing some variable's value. Consider our example of Figure 2. One possible sequence of constraint propagation steps is: (1) since the inputs to multiplier M_1 are known to be 2 and 3, X must be 6, (2) since the output of adder A_1 is 10 and one of its inputs X is 6, its other input must be $Y = 4$, (3) since the inputs to the multiplier M_2 are known, Y must be 6. At this point a discrepancy is detected at Y which we analyze in a moment. First, note that constraint propagation is not the same as simulation: a component constraint can be used to determine the output of a component as well as its inputs.

When a discrepancy is detected, MBD must determine which component constraints contributed to the discrepancy. An Assumption-Based Truth Maintenance System (de Kleer, 1986) will record, along with each propagated variable value, the set of all of the components used in the derivation. These sets are called the support environments of the propagated values. The ATMS will discard any support environments which are supersets of others for the same variable assignment. Such superset environments are logically redundant.

We use the notation $\langle x = v, \{\ldots\}\ldots\{\ldots\}\}\rangle$ to represent propagated values with their supporting environments. The propagation of $X = 6$ from $A = 3$ and $C = 2$ is represented as $\langle X = 6, \{\{M_1\}\}\rangle$ and corresponds to the logical sentence:

$$\neg AB(M_1) \rightarrow X = 6.$$

The later propagation of $F = 12$ from $X = 6$ and $Y = 6$ is represented as $\langle F = 12, \{\{A_1, M_1, M_2\}\}\rangle$.

Figure 7 is annotated with all the possible propagations and their supporting environments before F and G are measured. If F is measured to be 10, MBD immediately detects the discrepancy and records the conflict $\langle A_1, M_1, M_2\rangle$ corresponding to the supporting environment of $F = 12$. Constraint propagation continues producing the database described in Figure 8. The inference procedure will remove any supporting environment which contains a conflict because any such variable assignment will not hold in any diagnosis.

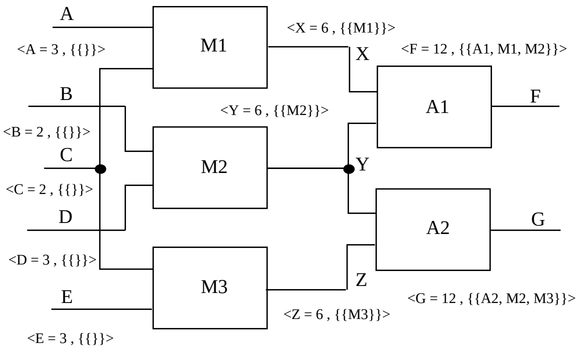

Fig. 7. Propagated values and supporting environments before any output observations.

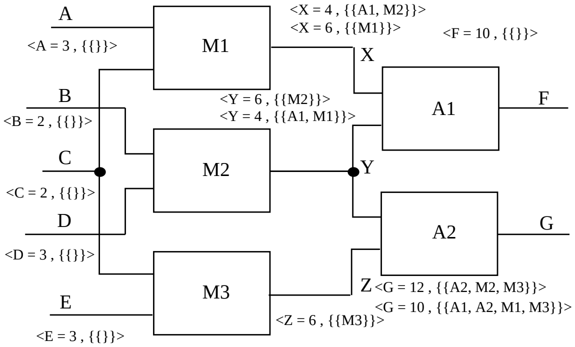

Fig. 8. Propagated values and supporting environments before measuring $G = 10$.

The resulting database is used to determine the values of variables in the different diagnoses. For example, Figure 8 shows that $X = 6$ in all diagnoses in which M_1 is unfaulted and $X = 4$ for all diagnoses where A_2 and M_2 are unfaulted. This database is then used to determine the probabilities of variable assignments. For example, $x = v$ in diagnosis D if one of the supporting environments of $x = v$ is a subset of the good components of D.

The implementation of a simple model-based diagnoser is described in (Forbus and de Kleer, 1992). The code for the system is available from `www.qrg.northwestern.edu/BPS/readme.html`

6.3 Focusing MBD

Even with the innovations of the previous two subsections, MBD engine performance is not adequate for most practical tasks. A task may still require an exponential number of minimal conflicts, minimal diagnoses and supporting environments. For example, Figure 9 shows how the number of minimal conflicts grows with the number of logic gates in a serial adder.

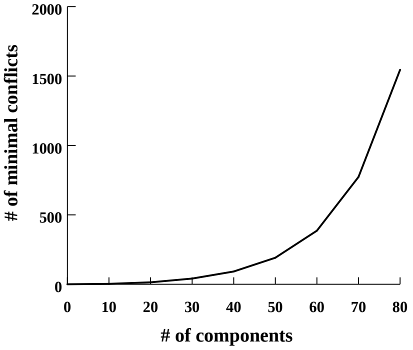

Fig. 9. Number of minimal conflicts vs. number of components.

For practical MBD tasks, the goal is to find the most probable diagnoses, not necessarily the minimal ones. Therefore, most MBD engines employ best-first search to discover diagnoses in decreasing posterior probability order. Typically, the search terminates when some number of leading diagnoses are identified. As additional measurements are collected, some of the leading diagnoses will be eliminated and the best-first search is restarted to find more leading diagnoses. For most diagnostic tasks this eliminates the need to explore an exponential number of possible diagnoses.

Using the leading diagnoses as a focusing mechanism, the constraint propagation can be limited to only draw inferences which are valid in some leading diagnosis (de Kleer, 1991). As a consequence the number of conflicts and supporting environments discovered will no longer be exponential. With these innovations, MBD engines can easily diagnose systems consisting of 10,000s of components. Although the resulting algorithm is incomplete, it may, on limited occasion, perform a suboptimal probe, but it will never exonerate a diagnosis incorrectly (i.e., it is sound).

7. INTRODUCING FAULT MODES

Knowledge of the ways a component fails can be of great help in diagnostic reasoning. For example, in the MBD framework presented thus far, a resistor is as likely to change its resistance as to become a current source. The former is

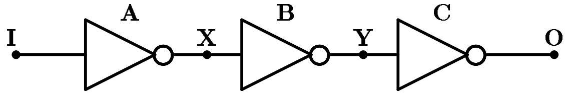

Fig. 10. Three sequential inverters.

dramatically more likely than the latter. Consider a simple 3 inverter circuit from the digital domain shown in Figure 10. Normally, the output of an inverter is 1 if the input is 0, and vice versa. Suppose the inverter can fail in two ways. The output may be stuck-at-0 (called SA0) regardless of input or may be stuck-at-1 (SA1). The fault model for an inverter can be written:

$$INVERTER(x) \wedge AB(x) \rightarrow [SA0(x) \vee SA1(x)],$$
$$SA0(x) \rightarrow out(x) = 0,$$
$$SA1(x) \rightarrow out(x) = 1.$$

The notion of mode can be extended to good modes as well, so components may have multiple good modes as well as bad modes. A diagnosis in this framework is an assignment of modes to components which is consistent with all the observations. The diagnosis framework of the previous sections can be seen as diagnosis with two modes, one good mode $\neg AB(x)$ and one bad mode $AB(x)$.

MBD approaches typically include an unknown failure mode for components. Intuitively, the unknown mode will always be consistent with unforeseen behavior, such as C shorting its input to its output ($out(C) = in(C)$). The fault modes are thus modeled as:

$$INVERTER(x) \wedge AB(x) \rightarrow$$
$$[SA0(x) \vee SA1(x) \vee U(x)],$$

where $SA1$ and $SA0$ are modeled as before and no behavior is specified for the unknown mode. The identification of which mode is more likely than another is left to the probabilistic overlay within which unknown modes are usually assigned very low prior probability. For example, suppose that inverters A and B tend to fail stuck-at-1 and C tends to fail stuck-at-0. The prior probabilities of failure might be as follows:

A	B	C
$p(G) = .99$	$p(G) = .99$	$p(G) = .99$
$p(S1) = .008$	$p(S1) = .008$	$p(S1) = .001$
$p(S0) = .001$	$p(S0) = .001$	$p(S0) = .008$
$p(U) = .001$	$p(U) = .001$	$p(U) = .001$

The MBD diagnostic process on the circuit of Figure 10 proceeds as follows. Given no observations, MBD finds a single leading diagnosis:

$$p([G(A), G(B), G(C)]) = .970.$$

Given a zero input ($I = 0$), MBD computes the following predictions and supporting environments.

$X = 0,\ \{S0(A)\}$
$X = 1,\ \{G(A)\}\{S1(A)\}$
$Y = 0,\ \{G(A), G(B)\}\{S1(A), G(B)\}\{S0(B)\}$
$Y = 1,\ \{S0(A), G(B)\}\ \{S1(B)\}$
$O = 0,\ \{S0(A), G(B), G(C)\}$
$\quad \{S1(B), G(C)\}\ \{S0(C)\}$
$O = 1,\ \{G(A), G(B), G(C)\}\{S1(A), G(B), G(C)\}$
$\quad \{S0(B), G(C)\}\ \{S1(C)\}$

Intuitively, the last line states that the output is one if either (1) all the components are good, (2) the first inverter is stuck-at-1 and the other two are good, (3) the second inverter is stuck-at-0, and the final inverter is good, or (4) the last inverter is stuck-at-1.

Applying the MBD probing strategy presented in Section 5 yields the following cost estimates:

$$\$(X) = .99,\ \ \$(Y) = .95,\ \ \$(O) = .91.$$

All these costs are high because there is no evidence that a fault exists. Since O has the lowest cost, it is measured next. Suppose it is measured to be 0. This results in four minimal conflicts (because each set of the minimal environments supporting $O = 1$ is now a conflict):

$$\neg G(A) \vee \neg G(B) \vee \neg G(C)\ ,\ \neg S1(A) \vee G(B) \vee \neg G(C),$$
$$\neg S0(B) \vee \neg G(C)\ ,\ \neg S1(C)$$

The leading diagnoses now are:

$$p([G(A), G(B), S0(C)]) = 0.426$$
$$p([G(A), S1(B), G(C)]) = 0.426$$
$$p([S0(A), G(B), G(C)]) = 0.053$$
$$p([G(A), G(B), U(C)]) = 0.027$$
$$p([U(A), G(B), G(C)]) = 0.027$$
$$p([G(A), G(C), U(B)]) = 0.027$$
$$Total = 0.986$$

Notice that although diagnosis $[S0(A), G(B), G(C)]$ has the same prior probability as the three diagnoses $[G(A), G(B), U(C)]$, $[U(A), G(B), G(C)]$ and $[G(A), G(C), U(B)]$, after the two measurements its probability is twice that of the others. This is because the three diagnoses predict no value of the output and hence their *posterior* probability is reduced by one-half by Bayes rule.

Given the leading diagnoses, the resultant probabilities of the component modes are:

	A	B	C
$p(G)$	.911	.538	.538
$p(S1)$	.007	.434	0
$p(S0)$	.054	.0005	.435
$p(U)$	.028	.027	.027

The table indicates that the major failure modes to consider are C stuck-at-0, and B stuck-at-1 and that all other faults are unlikely. The inference that A is unlikely to be faulted would not have been possible without the incorporation of fault models. In addition, the best place to measure next, namely Y, is now clear.

8. CONTINUOUS SYSTEMS

MBD can be used to perform useful diagnostic tasks on continuous systems through the use of qualitative models. A qualitative model captures a discrete abstraction of the behavior of a continuous system, which can be sufficient for performing diagnosis. In this section, a simple qualitative model is developed to provide the basic intuitions of qualitative modeling and reinforce the consistency-based diagnosis process.

In order to produce a qualitative model of a continuous system, the variables and constraints that model the system's behavior must discretized. Care must be taken to find a discretization that retains sufficient detail to enable diagnosis. Often this is accomplished by discretizing a residual of the system then capturing in SD a qualitative algebra that allows the residual to be propagated back to its source. Take as an example the diagnosis of the attitude control system (ACS) of the Deep Space 1 spacecraft, as performed by Livingstone (Williams and Nayak, 1996) during the Remote Agent Experiment (Bernard *et al.*, 1998). The ACS has a set of eight thrusters that are used to control the orientation of the spacecraft points by slowly rotating it about each of its three axes. Each thruster produces thrust in a single direction, and due to their orientation on the spacecraft, produce torque around its X, Y, or Z axes. The diagnostic problem is to determine which if any of the thrusters is not firing.

In order to diagnose the ACS, the measured error between the desired and actual orientation of the spacecraft in three dimensions was discretized into three variables that captured whether the pointing along each axis was high, low or nominal with respect to the desired pointing. Each of these discretizations required a small piece of software, referred to as a monitor, to convert from the real-valued sensor reading into the discrete space $\{low, nominal, high\}$. Figure 11 is a greatly simplified version of the ACS model used by Livingstone, reduced to a single axis. Lines 1 and 2 specify that if the thruster is not abnormal, it produces the expected, nominal amount of thrust, and that when it is abnormal, lower than expected thrust is produced. Lines 4-10 instantiate an ACS system for one axis, which is comprised of two thrusters, *tp* and *tn*. Thruster *tp* is mounted to cause positive rotation on the axis. Intuitively, line 7 specifies that the deviation in thrust for *tp* is the same as the deviation in torque. As *tn* is mounted in the negative direction, line 8 uses a qualitative negation operator, defined on lines 17 through 20, that specifies deviations in the thrust of *tn* cause a deviation in torque of the opposite sign. Line 9 specifies that the torque on the axis of the spacecraft is equal to the sum of the torques provided by each thruster. The sum is represented by the qualitative addition operator over the values $\{low, nominal, high\}$, defined on lines 12-15.

```
1   $THRUSTER(x) \wedge \neg AB(x) \rightarrow thrust(x) = nominal$
2   $THRUSTER(x) \wedge AB(x) \rightarrow thrust(x) = low$
3
4   $ACS(axis, tp, tn) \rightarrow$
5           $THRUSTER(tp) \quad \wedge$
6           $THRUSTER(tn) \quad \wedge$
7           $torque(tp) = thrust(tp) \quad \wedge$
8           $negate(torque(tn), thrust(tn)) \quad \wedge$
9           $torque(axis) = torque(tp) \oplus torque(tn)$
10          $rotation(axis) = torque(axis)$
11
12  $A = nominal \rightarrow A \oplus B = B$
13  $B = nominal \rightarrow A \oplus B = A$
14  $A = high \ \wedge B = high \rightarrow A \oplus B = high$
15  $A = low \ \wedge B = low \rightarrow A \oplus B = low$
16
17  $negate(A, B) \rightarrow$
18          $A = nominal \equiv B = nominal \quad \wedge$
19          $A = high \equiv B = low \quad \wedge$
20          $A = low \equiv B = high$
```

Fig. 11. A Qualitative Model for the ACS System

Note that if we assume $\neg AB(tp) \wedge \neg AB(tn)$ then by line 1, $thrust(tp) = thrust(tn) = nominal$. By lines 7 and 8, $torque(tp) = torque(tn) = nominal$, and by lines 9 and 12, $torque(axis) = nominal$. Thus by line 10

$$\neg AB(tp) \wedge \neg AB(tn) \rightarrow rotation(axis) = nominal$$

Therefore when $rotation(axis) = low$ is observed, the conflict generated is $AB(tp) \vee AB(tn)$. The MBD engine attempts to find the most likely consistent diagnosis. If the prior probabilities of $[tp]$ and $[tn]$ are equal, either may be explored first. Consider $[tn]$. By 2, 8, 17 and 20, $AB(tn) \rightarrow torque(tn) = high$. That is, because *tn* faces in the negative direction, it makes less contribution in the negative direction when failed. By 9 and 10, this would imply $rotation(axis) = high$, which is inconsistent. This diagnosis is therefore eliminated. The diagnosis with highest prior probability is now $[tp]$. It is considered and found to be consistent.

9. FURTHER TOPICS

All of the examples of this paper have been purposely simple in order to explain the fundamental concepts most directly. There are applications of MBD across the spectrum including, power grids, automotive electronics, ecological systems, analog circuits, digital circuits, chemical plants. MBD is a rich framework which can be expanded to handle many other aspects of diagnosis tasks which cannot be addressed in this paper. The following is a list of some the topics addressed in the MBD literature (the best sources are (Hamscher *et al.*, 1992) and the yearly proceedings of the International Workshop on Principles of Diagnosis).

- Alternative definitions of diagnosis (e.g., abductive diagnosis).
- Alternative methods for introducing probabilistic knowledge (e.g., Bayesian belief nets).
- Generation of FMEA (failure-mode and effects analysis) from models.
- Generation of diagnosis and repair manuals from models.
- Intermittent vs. non-intermittent faults.
- Repairing the system by replacing components or automatically managing redundancy.
- Changing the inputs of the system to obtain more diagnostic information.
- Minimizing the total overall cost of diagnosis (including customer down time, computation time, technician time, spare part usage).
- The automatic generation of software from the model.
- Automatic construction of models from design tools.
- Commercially available MBD tools such as from OCC'M Software (`www.occm.de`) and R.O.S.E. Informatik (`www.rose.de`).

Some of the most difficult issues to address in MBD engines are:

- Toleration of noise in observable variables
- Diagnosis of hybrid discrete/continuous systems including
    · Metric rather than discrete time
    · Autonomous transitions between modes governed via complex dynamics
    · Continuous degradation in addition to discrete failure modes
- Development or discovery of models adequate for diagnosis without excessive human engineering work.

These areas are extensively addressed in the FDI community and we hope this paper will help foster significant FDI-MBD cross fertilization on these topics.

REFERENCES

Bernard, D. E., G. A. Dorais, C. Fry, E. B. Gamble Jr., B. Kanefsky, J. Kurien, W. Millar, N. Muscettola, P. P. Nayak, B. Pell, K. Rajan, N. Rouquette, B. Smith and B. C. Williams (1998). Design of the remote agent experiment for spacecraft autonomy. In: *Proceedings of IEEE Aerospace*.

Cordier, M.O., P. Dague, M. Dumas, F. Levy, J. Montmain, M. Staroswiecki and L. Trave-Massuyes (2000). AI and automatic control approaches of model-based diagnosis: Links and underlying hypotheses. In: *Proceedings of the 4th IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes*. Budapest. pp. 274–279.

de Kleer, J. (1986). An assumption-based TMS. *Artificial Intelligence* **28**(2), 127–162.

de Kleer, J. (1991). Focusing on probable diagnoses. In: *Proc. 9th National Conf. on Artificial Intelligence*. Anaheim, CA. pp. 842–848.

de Kleer, J., A. Mackworth and R. Reiter (1992). Characterizing diagnoses and systems. **56**(2-3), 197–222.

de Kleer, J. and B. C. Williams (1987). Diagnosing multiple faults. *Artificial Intelligence* **32**(1), 97–130.

de Kleer, J. and B.C. Williams (1989). Diagnosis with behavioral modes. In: *Proc. 11th IJCAI*. Detroit. pp. 1324–1330.

Forbus, K. D. and J. de Kleer (1992). *Building Problem Solvers*. MIT Press. Cambridge, MA.

Hamscher, W. C., de Kleer, J. and Console, L., Eds.) (1992). *Readings in Model-based Diagnosis*. Morgan Kaufmann. San Mateo, Calif.

Kohavi, Z. (1978). *Switching and Finite Automata Theory*. McGraw-Hill.

Mauss, J., V. May and M. Tatar (2000). Towards model-based engineering: Failure analysis with mds. In: *Workshop on Knowledge-Based Systems for Model-Based Engineering, European Conference on AI (ECAI-2000*.

Reiter, R. (1987). A theory of diagnosis from first principles. *Artificial Intelligence* **32**(1), 57–96.

Sachenbacher, M., P. Struss and R. Weber (2000). Advances in design and implementation of obd functions for diesel injection systems based on a qualitative approach to diagnosis. In: *Proceedings of the SAE 2000 World Congress*. SAE.

Simon, L. and A. del Val (2001). Efficient consequence finding. In: *Proc. 17th Int. Joint Conf. on Artificial Intelligence*. Seattle, WA. pp. 359–365.

Tison, P. (1967). Generalized consensus theory and application to the minimization of boolean functions. *IEEE Transactions on Electronic Computers* **4**, 446–456.

Williams, B. C. and P. P. Nayak (1996). A model-based approach to reactive self-configuring systems. In: *Proc. 14th National Conf. on Artificial Intelligence*. pp. 971–978.