Solving Strong-Fault Diagnostic Models by Model Relaxation

Alexander Feldman\textsuperscript{1} and Gregory Provan\textsuperscript{2} and Arjan van Gemund\textsuperscript{1}
\textsuperscript{1}Delft University of Technology, Delft, The Netherlands
e-mail: \{a.b.feldman,a.j.c.vangemund\}@tudelft.nl
\textsuperscript{2}University College Cork, College Road, Cork, Ireland, e-mail: g.provan@cs.ucc.ie

Abstract
In Model-Based Diagnosis (MBD), the problem of computing a diagnosis in a strong-fault model (SFM) is computationally much harder than in a weak-fault model (WFM). For example, in propositional Horn models, computing the first minimal diagnosis in a weak-fault model (WFM) is in P but is \text{NP}-hard for strong-fault models. As a result, SFM problems of practical significance have not been studied in great depth within the MBD community. In this paper we describe an algorithm that renders the problem of computing a diagnosis in several important SFM subclasses no harder than a similar computation in a WFM. We propose an approach for efficiently computing minimal diagnoses for these subclasses of SFM that extends existing conflict-based algorithms like GDE (Sherlock) and CDA\textsuperscript{*}. Experiments on ISCAS85 combinational circuits show (1) inference speedups with CDA\textsuperscript{*}of up to a factor of 8, and (2) an average of 28\% reduction in the average conflict size, at the price of an extra low-polynomial-time consistency check for a candidate diagnosis.

1 Modeling for Diagnostic Inference
Model-based diagnosis (MBD), as formulated in terms of logic [Reiter, 1987], focuses on determining whether an assignment of failure status to a set of mode-variables is consistent with a system description and an observation (e.g., of sensor values). Hence, the diagnostic process consists of taking an observation OBS, and then inferring the failure-mode assignment (diagnosis) consistent with OBS.

Within MBD, two broad classes of model types have been specified: weak-fault models WFM [de Kleer et al., 1992] and strong-fault models SFM [Struss and Dressler, 1989]. Traditionally, WFM has been considered to be computationally simple, and SFM computationally hard. Weak-fault models describe a system only in terms of its normal (non-faulty) behaviour, whereas strong-fault models include a definition of some aspects of abnormal behaviour. Strong-fault models can avoid violating physical rules (cf. [Struss and Dressler, 1989]), but at the cost of increased complexity: moving from a binary-valued model with \( n \) components (which is adequate for weak-fault models) to one with \( m + 1 \) possible faulty values increases the maximum number of failure candidates from \( 2^n \) to \( (m + 1)^n \).

In terms of worst-case complexity, finding the first minimal diagnosis for a Horn model in WFM can be done in polynomial time, but finding the next minimal diagnosis is \text{NP}-complete [Friedrich et al., 1990]. In contrast, inference in strong-fault models entails computing kernel diagnoses [de Kleer et al., 1992], which is a \( \Sigma_2^P \)-hard task and is known to be computationally intensive in practice; for example, kernel diagnoses are given by the prime implicants of the minimal conflicts [de Kleer et al., 1992]. Further, the average case complexity of reasoning in WFM versus SFM increases from poly-time in \( n \) (WFM) to exponential in \( n \) (SFM) [de Kleer et al., 1992].

Given this intractability associated with inference using SFM, we show that, by closer examination of SFM, there is a spectrum of model types, and corresponding inference complexities. We identify two main categories of SFM, which we call literal-based SFM, ISFM, and function-based SFM, FSFM, and show that ISFM has the same properties (including inference complexity) as WFM, whereas SFM has the properties traditionally assigned to SFM.

This paper is the first detailed analysis of SFM, to our knowledge, which exploits model structure for computational advantage. It demonstrates the spectrum of fault modeling choices available to the system designer, and the computational implications such choices impose on the resulting diagnostic inference.

We propose a SFM algorithm that: (1) decomposes a strong-fault model into strong and weak sub-models; (2) computes diagnoses first in the “relaxed” weak sub-model; and then (3) discards any diagnosis which is not also a diagnosis in the strong sub-model. We identify classes of SFM in which the SFM diagnosis verification (step 3 above) can be done efficiently. Using ISCAS85 benchmark circuits, we have empirically demonstrated that: (1) our algorithm reduces the diagnosis computation time in CDA\textsuperscript{*}by up to a factor of 8; and (2) the average LTMS conflict size decreases (at the price of an extra consistency check, which has low-polynomial or better time-complexity for several classes of propositional strong-fault models).

2 Related Work
One of the key elements of our approach is decomposing a strong-fault model into strong and weak sub-models, and then
computing diagnoses first in the “relaxed” weak sub-model, and efficiently verifying if this diagnosis is also a diagnosis in the strong sub-model.

Močetić and Holzbaur [1994] have proposed a related approach: their diagnostic engine, IDA, computes diagnoses first with a structural model, which specifies the topological connections of components in the model. This model enables propagation of input values to components to output values, given that the component is OK, and is strictly less expressive than either weak- or strong-fault models. Given any diagnoses computed in a structural model, they are then verified by the weak fault-model, and finally the resulting minimal diagnoses are verified by the strong-fault model.

In contrast to Močetić and Holzbaur, who relax models into an abstract structural relation, we relax models by partitioning the model into weak and strong sub-models, and then verifying the diagnoses of the weak sub-model by the strong sub-model (exploiting the efficient satisfiability of particular classes of strong sub-model, such as ISFM). Furthermore, our work differs in that they make assumptions on the underlying MBD engines, incrementally updating the underlying conflict set, a step which is not necessary in our approach. In contrast to IDA, our relaxation scheme and algorithm are completely orthogonal to the diagnostic engine.

Our work also bears some relation to work on abstraction of propositional knowledge bases, e.g., [Selman and Kautz, 1996; del Val, 2000]. However, rather than relax a model to a Horn approximation, we relax a strong-fault diagnostic model to a more tractable sub-model, the weak-fault model.

This paper also characterises strong-fault models. In contrast to previous work discussing strong fault models, e.g., [Console and Torasso, 1991; Močetić and Holzbaur, 1994; Struss and Dressler, 1989], we propose tractable sub-classes of strong-fault models which occur in practice, such as stuck-at circuit models.

Maier and Sachenbacher [2008] propose a general abstraction approach within the framework of constraint optimization. This approach is orthogonal to ours, in that it adaptively abstracts constraints to control the memory requirements of message-passing algorithms within a tree-decomposition framework, whereas we abstract the model prior to any inference, independent of any particular inference algorithm. It is possible to combine these two approaches, which is a topic of future work.

Strong-fault model relaxation is related to hierarchical abstraction [Chittaro and Ranon, 2004]. Due to the bad worst-case computational complexity of diagnosis, hierarchical diagnosis is often necessary, and strong-fault model relaxation can be applied at each level of a hierarchical abstraction to further improve inference efficiency.

3 Diagnostic Model Taxonomy

We represent an artifact as a propositional Wff over a set V of variables.

**Definition 1 (Diagnostic System)**. A diagnostic system DS is defined as the triple DS = ⟨SD, COMPS, OBS⟩, where SD is a propositional theory over a set of variables V, COMPS ⊆ V, OBS ⊆ V. COMPS is the set of assumables, and OBS is the set of observables.

Throughout this paper we will assume that SD \[\not\models \bot\].

3.1 Diagnosis and Minimal Diagnosis

The traditional query in MBD computes terms of assumable variables which are explanations for the system description and an observation.

**Definition 2 (Health Assignment)**. Given a diagnostic system DS = ⟨SD, COMPS, OBS⟩, an observation \(\alpha\) over some variables in OBS, and a health assignment \(\omega\), \(\omega\) is a diagnosis iff SD \(\land \alpha \land \omega \not\models \bot\).

Traditionally, other authors [de Kleer and Williams, 1987] arrive at minimal diagnoses by computing a minimal hitting set of the minimal conflicts (broadly, minimal health assignments incompatible with the system description and the observation), while this paper makes no use of conflicts, hence the equivalent direct definition above.

In the MBD literature, a range of types of “preferred” diagnosis has been proposed. This turns the MBD problem into an optimization problem. We assume that we have a preference relation \(\preceq\) over diagnoses, which enables us to specify a minimal diagnosis with respect to \(\preceq\).

**Definition 3 (Minimal Diagnosis)**. A diagnosis \(\omega^\preceq\) is defined as minimal if no diagnosis \(\omega^\preceq\) exists such that \(\text{Lit}^- (\omega^\preceq) \subset \text{Lit}^- (\omega^\preceq)\).

This definition allows us to capture the two best-known definitions of minimality, namely subset-minimality (using order \(\preceq\)) and cardinality-minimality (using order \(\leq\) which minimizes the number of negative literals).

The set of all diagnoses/minimal-cardinality (MC) diagnoses of a system description SD and an observation \(\alpha\) is denoted as \(\Omega (SD, \alpha) / \Omega^\preceq (SD, \alpha)\), respectively. We are often interested in computing the size of \(|\Omega^\preceq (SD, \alpha)|\). We refer to the latter problem as counting all MC diagnoses of SD and \(\alpha\).

3.2 Classification of Diagnostic Fault Models

We now introduce our classification of fault models. Given a health variable \(h_i\) and an arbitrary Wff \(F_i\), normal behaviour for component \(i\) is denoted using the clause \(h_i \Rightarrow F_i\), and abnormal behaviour by \(\lnot h_i \Rightarrow F_i\). Hence a weak-fault model, as depicted in row 2 of Table 1, is given by \(\bigwedge_{i=1}^n (h_i \Rightarrow F_i)\); a strong-fault model consists of clauses denoting both normal and abnormal behaviour, i.e., \(\bigwedge_{i=1}^n (h_i \Rightarrow F_{i,1}) \land (\lnot h_i \Rightarrow F_{i,2})\).

We generalize the well-known class of stuck-at models using the strong-fault class ISFM, in which each clause denoting abnormal behaviour is given by \((\lnot h_i \Rightarrow l_i)\) for some literal \(l_i\); for example, this captures component \(i\) (with output \(l_i\) being stuck-at-0) as given by \([(h_i = \text{stuck-at-0}) \Rightarrow (l_i = 0)]\).

The class hISFM of Horn strong-fault models has the consequent Wff \(F_i\) restricted to Horn clauses. The class fISFM...
of functional strong-fault models allows the consequent Wff $F_1$ to take on any form. Finally, the class nISFM of negative-literal strong-fault models has the consequent Wff $F_2$ restricted to defining the negation of the normal behaviour of component $i$ given $h_i$.

4 SFM Algorithm

In this section we show that we can define an algorithm for specific classes of strong fault models which have complexity of the same class as weak fault models.

4.1 SFM Decomposability

We use the notion of model decomposability in this algorithm, which we introduce first.

**Proposition 1.** SD $\in$ SFM is decomposable, i.e., $SD = SD_w \wedge SD_s$ where $SD_w, SD_s$ are the weak and strong subsets of clauses, respectively, such that the subsets have no clauses in common, i.e., $SD_w \cap SD_s = \emptyset$.

**Proof.** If we have a consistent SD, there can be no clause of the form $h_i \land \neg h_i \Rightarrow F_i$, where $F_i$ is an arbitrary Wff; i.e., every clause can have only one of $h_i$ or $\neg h_i$ in it. Hence we must be able to partition the clauses into: SD$_w$ which consists of clauses of the form $h_i \Rightarrow F_i$, for $i = 1, \ldots, m$, and SD$_s$ which consists of clauses of the form $\neg h_i \Rightarrow F_i$, for $i = 1, \ldots, m$. As a consequence, we can decompose $SD = SD_w \cup SD_s$, such that $SD_w \cap SD_s = \emptyset$. $\square$

In order to make use of this decomposability property, we need to prove that the strong fault portion of the model SD$_s$ will constrain the diagnoses that are generated by the weak part of the model SD$_w$, since we compute the intersection of the diagnosis sets of the weak ($\Omega(SD_w, OBS)$) and strong ($\Omega(SD_s, OBS)$) sub-models, i.e., $\Omega(SD, OBS) = \Omega(SD_s, OBS) \cap \Omega(SD_w, OBS)$.

**Lemma 1.** For a strong fault model SD $\in$ SFM which is decomposable such that $SD = SD_s \wedge SD_w$, we must have $\Omega(SD, \alpha) = \Omega(SD_w, \alpha) \setminus \Omega(SD_s, \alpha)$.

**Proof.** Since we are using a monotonic propositional logic, adding extra clauses to any formula $F$ will reduce the number of logical models (diagnoses) of $F$. It is easy to show that $\Omega(SD_w, \alpha) \supseteq \Omega(SD_s, \alpha)$. The diagnoses of $SD_s$ alone are given by $\Omega(SD_s, \alpha)$, and diagnoses excluded by $SD_s$ alone are given by $\Omega(SD_s, \alpha)$. Hence by adding $SD_s$ to $SD_w$ we obtain $\Omega(SD, \alpha) = \Omega(SD_w, \alpha) \setminus \Omega(SD_s, \alpha)$. $\square$

### 4.2 SFM Algorithm

Algorithm 1 shows how SFM decomposability can be used to extend existing MBD engines which can benefit from reasoning in WFM.

**Algorithm 1 SFM Decomposition Algorithm**

1: function DIAGNOSE(DS, $\alpha$) returns a set of diagnoses

   **inputs:** DS = (SD, COMP, OBS), diag. system $\alpha$, term, observation

   **local variables:** $\omega$, term, diagnosis

   $z$, term, internal variables

   SD$_w$, SD$_s$, system descriptions

   $\Omega$, result, set of diagnoses

2: SD$_w$ $\leftarrow$ NOMINALBEHAVIOR(SD)

3: SD$_s$ $\leftarrow$ FAULTYBEHAVIOR(SD)

4: while MOREDIAGNOSES?(SD$_w$, $\alpha$) do

5:  $\omega$ $\leftarrow$ NEXTDIAGNOSE(SD$_w$, $\alpha$)

6:  $U$ $\leftarrow$ COMPUTEINTERNALS(SD$_w$, $\alpha$, $\omega$)

7:  if SD$_s$ $\land$ $\alpha$ $\land$ $\omega$ $\land$ $U$ $\not\models$ then

8:     $\Omega$ $\leftarrow$ $\Omega$ $\cup$ \{ $\omega$ \}

9: end if

10: end while

11: return $\Omega$

12: end function

Note that Alg. 1 is independent of the diagnostic inference engine $\Xi$. In lines 2-3, we decompose the model; in line 5 we compute a diagnosis $\omega$ using the weak portion SD$_w$ of the model using $\Xi$. Then, line 7 determines if $\omega$ is consistent with the strong portion of model, SD$_s$. Our SFM

<table>
<thead>
<tr>
<th>Notation</th>
<th>Model Class</th>
<th>System Description</th>
<th>Restrictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>abduction models</td>
<td>propositional Wff</td>
<td>none of $h_i$ appears in $F_j$, $h_i \in$ COMP, $1 \leq i, j \leq n$</td>
</tr>
<tr>
<td>WFM</td>
<td>weak-fault models</td>
<td>$\bigwedge_{i=1}^n (h_i \Rightarrow F_i)$</td>
<td>none of $h_i$ appears in $F_j$, $h_i \in$ COMP, $1 \leq i, j \leq n$</td>
</tr>
<tr>
<td>SFM</td>
<td>strong-fault models</td>
<td>SFM = lSFM $\cup$ hSFM $\cup$ fSFM</td>
<td>none of $h_i$ appears in $F_j$, $h_i \in$ COMP, $1 \leq i, j \leq n$</td>
</tr>
<tr>
<td>lSFM</td>
<td>literal-based strong-fault models</td>
<td>$\bigwedge_{i=1}^n (h_i \Rightarrow F_i)$</td>
<td>none of $h_i$ appears in $F_j$, $h_i \in$ COMP, $1 \leq i, j \leq n$</td>
</tr>
<tr>
<td>hSFM</td>
<td>Horn strong-fault models</td>
<td>$\bigwedge_{i=1}^n (h_i \Rightarrow F_i)$</td>
<td>none of $h_i$ appears in $F_j$, $h_i \in$ COMP, $1 \leq i, j \leq n$</td>
</tr>
<tr>
<td>fSFM</td>
<td>functional strong-fault models</td>
<td>$\bigwedge_{i=1}^n (h_i \Rightarrow F_i)$</td>
<td>none of $h_i$ appears in $F_j$, $h_i \in$ COMP, $1 \leq i, j \leq n$</td>
</tr>
<tr>
<td>nISFM</td>
<td>negative literal strong-fault models</td>
<td>$\bigwedge_{i=1}^n (h_i \Rightarrow F_i)$</td>
<td>none of $h_i$ appears in $F_j$, $h_i \in$ COMP, $1 \leq i, j \leq n$</td>
</tr>
</tbody>
</table>

Table 1: Model classification
algorithm relies on being able to assign values in function \textsc{ComputeInternals} to all internal variables, the correctness of which we show in Lemma 3.

The complexity of Diagnose(DS, \alpha) depends on the complexity of three steps:

1. Computing a diagnosis in a weak fault model using a diagnostic oracle \Xi;
2. Computing values of internal variables \(U = V \setminus (\text{OBS} \cup \text{COMPS})\);
3. Computing a diagnosis in the strong portion of the fault model, SD_s.

Our algorithm is independent of the diagnostic oracle \Xi, so we state the worst-case complexity here, and do not focus on the average-case complexity of any particular inference algorithm.

The complexity of computing a diagnosis in a weak fault model SD \in WFM (e.g., in the weak portion SD_w of the fault model) is as follows:

**Lemma 2.** Given \(SD \in WFM\), an observation \alpha, the complexity of computing the first diagnosis \(\omega(\cdot, \alpha)\) is \(\Sigma^p_2\)-complete, and of computing subsequent diagnoses is \(\Sigma^p_2\)-complete.

**Proof.** For the class of propositional Horn abduction problems, Bylander et al. [1991] show that computing the first diagnosis \(\omega(\cdot, \alpha)\) is polynomial in \(|\text{COMPS}| + |\text{OBS}|\), and of computing subsequent diagnoses is \(\Sigma^p_1\)-complete. Since we now deal with general propositional problems, the respective complexities will be up one level of the polynomial hierarchy, i.e., computing the first diagnosis \(\omega(\cdot, \alpha)\) is \(\Sigma^p_2\)-complete, and of computing subsequent diagnoses is \(\Sigma^p_2\)-complete. Further, these results concur with appropriate formulations in [Nordh and Zanuttini, 2008].

The complexity of computing the internal variables is as follows:

**Lemma 3.** Given \(SD \in WFM\), an observation \alpha assigning values to all variables in OBS, and a diagnosis \(\omega\), we can assign values to all internal variables \(U = V \setminus (\text{OBS} \cup \text{COMPS})\) in time \(O(|\text{COMPS}| \cdot |V|)\).

**Proof (Sketch).** We first show the correctness of this procedure, and then show its complexity.

**Correctness:** Let OBS = IN \cup OUT. For \(SD \in WFM\) and \alpha, diagnosis \(\omega\) assigns all \(h_i\) for the clauses \(C_i\). For each component \(i\) that has all inputs in IN, we can compute its outputs given \(h_i\). Here we have to distinguish (1) \(h_i\) and (2) \(\neg h_i\). In (1) the output of each gate can be computed directly from the inputs and the assumption that the gate is functioning correctly. In (2) the output of the gate is opposite to the one of a correctly functioning gate, otherwise the diagnostic inference cannot deduce \(\neg h_i\). If the circuit is connected, which we assume is the case, we can recursively compute the outputs of all downstream components, using induction on COMPS and the connectivity assumption.

**Complexity:** In the worst case every component has \(O(|V|)\) outputs, each of which can be determined in \(O(1)\) time, hence a total of \(O(|\text{COMPS}| \cdot |V|)\) computations.

For \(\text{nlSFM}\), we prove the following Lemma to be used later.

**Lemma 4.** The strong part SD_s of SD \in \text{nlSFM} is isomorphic to SD \in WFM.

**Proof.** We show this isomorphism by rewriting SD_s, so that it has the form of SD_w. SD_s has the form \(\neg h_i \Rightarrow \neg F_i\), and SD_w has the form \(h_i \Rightarrow F_i\). If we rename the literals in SD_s by replacing each literal \(v \in V\) such that \(\neg v_i\) is replaced with \(\bar{v}_i\), we obtain SD_s with the form \(h_i \Rightarrow F_i\). The description for the strong part is now isomorphic to the description for the weak part, up to variable renaming.

The complexity of computing a diagnosis in the strong portion SD_s of the fault model depends on the class of the model. We focus on the following strong fault model classes: ISFM, nlSFM, hSFM, fSFM.

**Lemma 5.** Given the strong portion SD_s of SD \in ISFM, an observation \alpha assigning values to all variables in OBS, diagnosis \(\omega(\cdot, \alpha)\) for the weak portion SD_w of SD \in ISFM, and values for internal variables \(U = V \setminus (\text{OBS} \cup \text{COMPS})\), the worst-case complexity of the consistency check SD_s \& \alpha \land \omega \land U \models \bot is given by:

- \(O(|V|)\) for SD \in ISFM;
- \(O(|V|)\) for SD \in hSFM;
- \(\Sigma^p_1\)-complete for the first diagnosis for SD \in nlSFM;
- \(\Sigma^p_2\)-complete for SD \in fSFM.

**Proof.** For SD \in ISFM, the strong part of the system description SD_s is a 2SAT Boolean function, since it has the form \(\wedge\neg h_i \Rightarrow F_i\Rightarrow F_i\), where \(F_i\) is any CNF Wff that does not include \(h_i\), for \(i = 1, ..., n\). Using Lemma 4, we can show that this task has the same complexity as computing a weak-fault diagnosis.

For SD \in hSFM, SD_s has the form \(\neg h_i \Rightarrow F_i\). Given we have assignments to all the \(h_i\), we need to check the consistency of a set of Horn formulae each of which is set to \(t\) or \(f\). Since consistency checking of a set of Horn formulae is a linear-time operation, we can evaluate the consistency of SD_s in linear time.

For SD \in nlSFM, the strong part of the system description SD_s has the form \(\wedge\neg h_i \Rightarrow (\neg o_i \Leftrightarrow C_i)\), where \(C_i\) is any CNF Wff that does not include \(o_i\) or \(h_i\), for \(i = 1, ..., n\). Using Lemma 4, we can show that this task has the same complexity as computing a weak-fault diagnosis.

For SD \in fSFM (the case of general clauses), the complexity reverts to the worst-case result of \(\Sigma^p_2\)-completeness.

5 Experimental Results

Our approach applies to any domain in which models can be defined as in Table 1. We empirically demonstrate our approach using the well-known ISCAS85 circuits [Brglez and Fujiwara, 1985]. In addition to the original ISCAS85 models, we have performed cone reductions as described by Siddiqi and Huang [2007] and de Kleer [2008].

Table 2 gives an overview of all models (\(V\) and \(C\) denote the total number of variables and number of clauses, respectively). For each circuit we have generated 100 non-masking
Next we have repeated the counting of MC diagnoses in double-faults and counted the number of minimal-cardinality models. For each circuit and observation we computed the average conflict size from the “all nominal” candidate (SD ∧ α leads to a double fault, hence assuming each component is healthy leads to a conflict) and from all single-fault candidates. The WFM columns show these values averaged over all observations. The remaining three columns show the ratio of the average conflict sizes in WFM to the respective SIFM subclasses.

<table>
<thead>
<tr>
<th>id</th>
<th>original</th>
<th>reduced</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[OBS]</td>
<td>[COMPS]</td>
</tr>
<tr>
<td>c432</td>
<td>43</td>
<td>160</td>
</tr>
<tr>
<td>c499</td>
<td>73</td>
<td>202</td>
</tr>
<tr>
<td>c880</td>
<td>86</td>
<td>383</td>
</tr>
<tr>
<td>c1355</td>
<td>73</td>
<td>546</td>
</tr>
<tr>
<td>c1908</td>
<td>58</td>
<td>880</td>
</tr>
<tr>
<td>c2670</td>
<td>373</td>
<td>1193</td>
</tr>
<tr>
<td>c3540</td>
<td>72</td>
<td>1669</td>
</tr>
<tr>
<td>c5315</td>
<td>301</td>
<td>2307</td>
</tr>
<tr>
<td>c6288</td>
<td>64</td>
<td>2416</td>
</tr>
<tr>
<td>c7552</td>
<td>315</td>
<td>3512</td>
</tr>
</tbody>
</table>

Table 2: niSFIM ISCAS85 models

Table 3: Number of MC diagnoses (niSFIM)

Next we have repeated the counting of MC diagnoses in stuck-at-0 and stuck-at-1 models (SD ∈ SIFM). The results are shown in Table 4 (data for stuck-at-1 are very similar).

With SIFM, there are observations α such that SD ∧ α = ⊥. The number of these “inconsistent” observations are given in the θ columns of Table 4. Note that, in practice, θ increases with the cardinality of the MC diagnoses.

Next we studied the effect of the fault modes on the average conflict size computed by a Logic-Based Truth Maintenance System (LTMS) [Forbus and de Kleer, 1993]. The conflict size is very important for algorithms like GDE [de Kleer and Williams, 1987] and CDA∗ [Williams and Ragno, 2007]. The LTMS conflict sizes for full models are shown in Table 5. We have measured similar reduction in the conflict size for reduced models. For each circuit and observation we computed the average conflict size from the “all nominal” candidate (SD ∧ α leads to a double fault, hence assuming each component is healthy leads to a conflict) and from all single-fault candidates. The WFM columns show these values averaged over all observations. The remaining three columns show the ratio of the average conflict sizes in WFM to the respective SIFM subclasses.

Table 5: Average LTMS conflict size

Interestingly, the savings in the average conflict size (between 30% and 36%) are biggest in c6288, which is considered the most difficult ISCAS85 circuit in MBD.

Finally, we studied the effect of Alg. 1 on CDA∗, whose average-case complexity is governed by the following two NP-hard problems, which must be solved to compute minimal diagnoses, given a set of (non-minimal) conflicts: (1) compute the set of minimal conflicts by using, for example, directed resolution, and (2) compute the Minimal Hitting Sets (MHS) of all minimal conflicts [Reiter, 1987]. In this experiment we measured the time for computing the first 10 diagnoses in SIFM with (1) CDA∗ and (2) Alg. 1 (i.e., CDA∗ with WFM models and then discarding part of the WFM diagnoses after fast consistency check). Table 6 demonstrates speed-ups of up to 750%. Note that our CDA∗ implementation is pretty limited in solving only “easier” diagnostic problems, and the advantage of model relaxation would be more visible with problems leading to diagnoses of higher cardinalities (more conflicts).

Table 6 shows that Alg. 1 leads to a considerable diagnostic speedup (up to a factor of 8) and that the speedup increases...
with the model size. The speedup is better observed with the non-reduced models. We expect even better speedup for diagnosis instances leading to a higher number of conflicts (and respectively higher MC diagnosis cardinality).

6 Conclusions

We have provided a classification of strong-fault diagnostics models, and have shown some computational properties of key sub-classes of such models. In particular, by decomposing a strong-fault model into disjoint strong and weak sub-models, we have shown a relaxation algorithm which improves the efficiency of conflict-based inference algorithms by computing first the diagnoses for the weak sub-model, $\Omega(SD_w, \alpha)$, and then removing any diagnoses in $\Omega(SD_w, \alpha)$ inconsistent with the strong sub-model, which can be performed in low-polynomial or better time for several classes of propositional strong-fault models.

Our method is complementary to algorithms (such as conflict-based algorithms), which, on average, work faster with WFM models. Using ISCAS85 circuits, we have empirically demonstrated that: (1) a large portion of the WFM models are also diagnoses in the corresponding SFM models; (2) the average LTMS conflict size decreases; and (3) the diagnosis computation time in CDA* decreases by up to a factor of 8. Our method gives best results with the “difficult” c6288 circuit, and we conjecture that its speedups will increase with the complexity of diagnostic inference.

In the future we plan to extend our methods beyond propositional logic. One straightforward extension of our strong-fault model classification is to many-valued logic, where components may have more than one nominal state; similar relaxation (or strengthening) techniques are also applicable to temporal logic.

References


