Computing Observation Vectors for Max-Fault Min-Cardinality Diagnoses

Alexander Feldman
Delft University of Technology
Mekelweg 4, 2628 CD, Delft
The Netherlands
Tel.: +31 15 2781935
email: a.b.feldman@tudelft.nl

Gregory Provan
University College Cork
College Road, Cork, Ireland
Tel: +353 21 4901816
email: g.provan@cs.ucc.ie

Arjan van Gemund
Delft University of Technology
Mekelweg 4, 2628 CD, Delft
The Netherlands
Tel.: +31 15 2781935
email: a.j.c.vangemund@tudelft.nl

Abstract

Model-Based Diagnosis (MBD) typically focuses on diagnoses, minimal under some minimality criterion, e.g., the minimal-cardinality set of faulty components that explain an observation α. However, for different α there may be minimal-cardinality diagnoses of differing cardinalities, and several applications (such as test pattern generation and benchmark model analysis) need to identify the α leading to the max-cardinality diagnosis amongst them. We denote this problem as a Max-Fault Min-Cardinality (MFMC) problem. This paper considers the generation of observations that lead to MFMC diagnoses. We present a near-optimal, stochastic algorithm, called MIRANDA (Max-fault mIn-caRdinAlity observatioN Deduction Algorithm), that computes MFMC observations. Compared to optimal, deterministic approaches such as ATPG, the algorithm has very low cost, allowing us to generate observations corresponding to high-cardinality faults. Experiments show that MIRANDA delivers optimal results on the 74XXX circuits, as well as good MFMC cardinality estimates on the larger ISCAS85 circuits.

Introduction

The problem of computing minimal-cardinality diagnoses, given an observation and a system description, is central to Model-Based Diagnosis (de Kleer and Williams 1987). In this paper we consider the "inverse" problem of computing an observation that simultaneously isolates k faulty components. These observations are useful in system testing and benchmarking of multiple-fault diagnostic techniques. Computing observations (in particular inputs) that distinguish a single failing component ($k = 1$) is studied by Automatic Test Pattern Generation (ATPG) and dates back to the D-algorithm (Roth 1966). The goal of ATPG is to compute a *sequence* of test vectors that can detect every possible single fault in a device. Single-fault ATPG has been extended to finding observation vectors leading to double faults (Hughes 1988) and to multiple faults (Kubiak and Fuchs 1991). These approaches have several drawbacks, including: (1) they do not determine the maximum possible value of k (2) they suffer from very high computational complexity, and (3) they severely limit the class of system abstractions by imposing various model restrictions.

Few papers have proposed algorithms computing observation vectors that distinguish the *maximum* number of failing components in a system (Abramovici 1981). The GUIDEDPROBE algorithm in the latter paper relies on probing to achieve the maximal fault resolution for a fixed test T. The author of this algorithm has a different goal, i.e., achieving maximal resolution by minimizing the number of probes, and proposes essentially a sequential algorithm. This is very different from MBD approaches, which try to solve the multiple-fault problem with only one observation.

To the best of our knowledge, we are the first to formally state the problem and significance of finding MFMC observation vectors, and then to define an algorithm that is able to approximate such a computationally difficult problem. Our method is based on a greedy stochastic search algorithm, called MIRANDA (Max-fault mIn-caRdinAlity observatioN Deduction Algorithm), and uses an MBD oracle for computing minimal-cardinality diagnoses. The algorithm is greedy in that it monotonically exploits part of the problem search space. The performance of our method is determined by the efficiency of the underlying MBD engine; i.e., it is efficient with a fast (usually incomplete) procedure for computing minimal-diagnoses.

One advantage of MIRANDA over related k-fault ATPG algorithms is that it uncovers the maximum value of k. Furthermore, it does not impose any limitations on the model (e.g., it neither requires no stuck-at modes nor assumes unlimited observability). This makes our approach applicable not only to system testing but also to MBD benchmarking and to a wider range of Model-Based Reasoning (MBR) problems, such as optimal sensor placement (Console, Picardi, and Ribaudo 2000), active testing, etc. In this paper the MFMC algorithm is applied to MBD benchmarking (Provan and Wang 2007), but it can be applied to compute a set of MFMC test vectors covering all components in a system.

We have evaluated the performance of MIRANDA using the ISCAS85 benchmark extended with 4 smaller circuits from the 74XXX family. For the 74XXX circuits we have been able to exactly compute all MFMC observation vectors. Since deterministic MBD algorithms cannot compute the high fault-cardinalities associated with MFMC vectors for the ISCAS85 circuits, we have used a stochastic MBD oracle (Feldman, Provan, and van Gemund 2007).

A summary of our contributions follows. This paper introduces the MFMC problem and an algorithm for computing MFMC observation vectors. We empirically analyze

the algorithm on a number of diagnostic models from the ISCAS85 and 74XXX benchmarks. We also provide an analytical method for estimating MFMC fault cardinalities.

This paper is organized as follows. The next two sections define the basic MFMC framework and MFMC algorithm, respectively. Finally, we show empirical results of testing the MFMC algorithm on a family of combinatorial circuits.

Technical Background

This paper uses the traditional diagnostic definitions (de Kleer, Mackworth, and Reiter 1992), except that we use propositional logic terms (conjunctions of literals) instead of sets of failing components.

Central to MBD, a *model* of an artifact is represented as a propositional **Wff** over some set of variables. Discerning two subsets of these variables as *assumable* and *observable*[1] variables gives us a diagnostic system.

Definition 1 (Diagnostic System)**.** A diagnostic system DS is defined as the triple $\mathrm{DS} = \langle \mathrm{SD}, \mathrm{COMPS}, \mathrm{OBS} \rangle$, where SD is a propositional theory over a set of variables V, $\mathrm{COMPS} \subseteq V$, $\mathrm{OBS} \subseteq V$, COMPS is the set of assumables, and OBS is the set of observables.

Throughout this paper we assume that $\mathrm{OBS} \cap \mathrm{COMPS} = \emptyset$ and $\mathrm{SD} \not\models \bot$. Although not necessary for MBD applications, a partitioning of OBS into an input set IN and an output set OUT ($\mathrm{OBS} = \mathrm{IN} \cup \mathrm{OUT}$ and $\mathrm{IN} \cap \mathrm{OUT} = \emptyset$) is convenient, familiar from ATPG, and allows an easier presentation of the MFMC algorithm.

A Running Example

We will use the Boolean circuit shown in Fig. 1 as a running example for illustrating all the notions and algorithm in this paper. The subtractor, shown there, consists of seven components: an inverter, two or-gates, two xor-gates, and two and-gates. The expression $h \Rightarrow (o \Leftrightarrow \neg i)$ models the normative (healthy) behavior of an inverter, where the variables i, o, and h represent input, output and health respectively. Similarly, an and-gate is modeled as $h \Rightarrow (o \Leftrightarrow i_1 \wedge i_2)$ and an or-gate by $h \Rightarrow (o \Leftrightarrow i_1 \vee i_2)$. Finally, an xor-gate is specified as $h \Rightarrow [o \Leftrightarrow \neg (i_1 \Leftrightarrow i_2)]$.

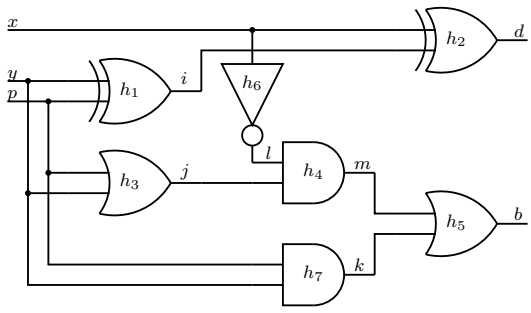

Figure 1: A subtractor circuit

[1] In the MBD literature the assumable variables are also referred to as "component", "failure-mode", or "health" variables. Observable variables are also called "measurable", or "control" variables.

The above propositional formulae are copied for each gate in Fig. 1 and their variables renamed in such a way as to properly connect the circuit and disambiguate the assumables, thus obtaining a propositional formula for the Boolean subtractor, given by:

$$\mathrm{SD} = \begin{cases} h_1 \Rightarrow [i \Leftrightarrow \neg (y \Leftrightarrow p)] \\ h_2 \Rightarrow [d \Leftrightarrow \neg (x \Leftrightarrow i)] \\ h_3 \Rightarrow (j \Leftrightarrow y \vee p) \\ h_4 \Rightarrow (m \Leftrightarrow l \wedge j) \\ h_5 \Rightarrow (b \Leftrightarrow m \vee k) \\ h_6 \Rightarrow (x \Leftrightarrow \neg l) \\ h_7 \Rightarrow (k \Leftrightarrow y \wedge p) \end{cases}$$

The assumable variables $\mathrm{COMPS} = \{h_1, h_2, \ldots, h_7\}$, the observable variables $\mathrm{OBS} = \{x, y, p, d, b\}$, the inputs are $\mathrm{IN} = \{x, y, p\}$, and the outputs $\mathrm{OUT} = \{b, d\}$.

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 $\mathrm{DS} = \langle \mathrm{SD}, \mathrm{COMPS}, \mathrm{OBS} \rangle$, an assignment HA to all variables in COMPS is defined as a health assignment.

A health assignment HA is a conjunction of propositional literals. In some cases it is convenient to use the set of negative or positive literals in HA. These two sets are denoted as $Lit^{-}(\mathrm{HA})$ and $Lit^{+}(\mathrm{HA})$, respectively.

In our example, the "all nominal" assignment is $\mathrm{HA}_1 = h_1 \wedge h_2 \wedge \cdots \wedge h_7$. The health assignment $\mathrm{HA}_2 = h_1 \wedge h_2 \wedge h_3 \wedge \neg h_4 \wedge h_5 \wedge h_6 \wedge \neg h_7$ means that the two and-gates from Fig. 1 are malfunctioning. What follows is a formal definition of consistency-based diagnosis.

Definition 3 (Diagnosis)**.** Given a diagnostic system $\mathrm{DS} = \langle \mathrm{SD}, \mathrm{COMPS}, \mathrm{OBS} \rangle$, an observation α over some variables in OBS, and a health assignment ω, ω is a diagnosis iff $\mathrm{SD} \wedge \alpha \wedge \omega \not\models \bot$.

There is a total of 96 possible diagnoses given SD and an observation $\alpha_1 = x \wedge y \wedge p \wedge b \wedge \neg d$. Example diagnoses are $\omega_1 = \neg h_1 \wedge h_2 \wedge h_3 \wedge \cdots \wedge h_7$ and $\omega_2 = h_1 \wedge \neg h_2 \wedge h_3 \wedge h_4 \wedge \cdots \wedge h_7$.

In the MBD literature, a range of types of "preferred" diagnosis has been proposed. This turns the MBD problem into an optimization problem. In the following definition we consider the common subset-ordering.

Definition 4 (Minimal Diagnosis)**.** A diagnosis ω is minimal if no diagnosis ω' exists such that $Lit^{-}(\omega') \subset Lit^{-}(\omega)$.

Traditionally, other authors (de Kleer and Williams 1987) arrive at minimal diagnosis 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.

For the model SD of the circuit shown in Fig. 1, and an observation $\alpha_2 = \neg x \wedge y \wedge p \wedge \neg b \wedge d$, there are 8 minimal and 61 non-minimal diagnoses. In this example, two of the minimal diagnoses are $\omega_3 = \neg h_1 \wedge h_2 \wedge h_3 \wedge h_4 \wedge \neg h_5 \wedge$

$h_6 \wedge h_7$ and $\omega_4 = \neg h_1 \wedge h_2 \wedge h_3 \wedge h_4 \wedge h_5 \wedge \neg h_6 \wedge \neg h_7$. The diagnosis $\omega_5 = \neg h_1 \wedge \neg h_2 \wedge h_3 \wedge h_4 \wedge \neg h_5 \wedge h_6 \wedge h_7$ is non-minimal as the negative literals in ω_3 form a subset of the negative literals in ω_5.

Definition 5 (Cardinality of a Diagnosis)**.** The cardinality of a diagnosis, denoted as $|\omega|$, is defined as the number of negative literals in ω.

Diagnosis cardinality gives us another partial ordering: a diagnosis is defined as *minimal-cardinality* iff it minimizes its number of negative literals.

The cardinality of a minimal cardinality diagnosis computed from a system description SD and an observation α is denoted as $MinCard(\mathrm{SD} \wedge \alpha)$. For our example model SD and an observation $\alpha_3 = x \wedge y \wedge p \wedge \neg b \wedge \neg d$, it follows that $MinCard(\mathrm{SD} \wedge \alpha_3) = 2$. In this particular case, all minimal diagnoses are also minimal-cardinality diagnoses.

A minimal-cardinality diagnosis is a minimal diagnosis, but the opposite does not hold. There are minimal diagnoses which are not minimal-cardinality diagnoses. Consider the example model SD, the observation α_2, and the two resulting minimal diagnoses ω_3 and ω_4 given earlier in this section. From the two diagnoses, only ω_3 is a minimal-cardinality diagnosis.

Keeping the model SD fixed, a different observation α may lead to a different $MinCard(\mathrm{SD} \wedge \alpha)$. This leads to our main definition.

Definition 6 (MFMC Observation)**.** Given a diagnostic system $\mathrm{DS} = \langle \mathrm{SD}, \mathrm{COMPS}, \mathrm{OBS} \rangle$, an observation α is defined as Max-Fault Min-Cardinality (MFMC) observation, iff ω is a minimal-cardinality diagnosis of $\mathrm{SD} \wedge \alpha$ and $|\omega|$ is maximized.

In addition to an MFMC observation, we also refer to an MFMC diagnosis ω of a model SD, which refers to any of the diagnoses entailed by an MFMC observation α. The cardinality of this diagnosis is denoted as $MFMC(\mathrm{SD})$ and, next to the associated MFMC observations, this is a key model property we seek to compute.

MFMC Algorithm

A naïve approach to compute $MFMC(\mathrm{SD})$ is to consider an exhaustive algorithm. Such an algorithm would enumerate all the $2^{|\mathrm{OBS}|}$ instantiations of the variables in OBS; one can easily show that only an assignments to *all* variables in OBS can be an MFMC observation vector as the MFMC problem is monotonic in respect to partial observations. For each full instantiation α, an MBD oracle computes the associated minimal fault cardinality.

Taking this exhaustive approach in our running example, we compute that $MFMC(\mathrm{SD}) = 2$ and that there is a total of 9 observation vectors discerning a minimal-cardinality diagnosis of 2 faults (α_2 and α_3 from the preceding section are examples of such observation vectors). From all the 32 possible observation vectors, there are 7, 16, and 9 observation vectors leading to a nominal, single-fault, and double-fault minimal-cardinality diagnosis, respectively.

Of course, such an exhaustive algorithm is computationally infeasible. We propose a stochastic method that trades optimality for a huge speedup, allowing very-high-k observations to be computed for very large circuits. Despite the inherent suboptimality of the stochastic approach, we will see in the experimental section of this paper that, for smaller circuits from the 74XXX family, using MIRANDA results in optimal observation vectors. The success of our stochastic approach is that, as we will see in the experimental section, landscapes of typical MFMC search problems have many optima which are close or equal to the global optimum.

Alg. 1 assumes that an "all-healthy" mode of all assumable variables allows an input assignment to be propagated to all outputs. This is typical for health-models of digital circuits and for diagnosis problems.

Algorithm 1 A greedy stochastic algorithm for generation of MFMC observation vectors

```
1:  function CLIMB(DS, IN, OUT, N) returns a term
        inputs: DS, a diagnostic system
                    DS = ⟨SD, COMPS, OBS⟩
                IN, OUT, variable sets
                IN ∪ OUT = OBS, IN ∩ OUT = ∅
                N, an integer, number of runs
        local variables: β, γ, γ', ω, R, terms
                         n, q, integers
                         l, a literal
2:      n ← 0
3:      q ← 0
4:      repeat
5:          β ← RANDOMINPUTS(IN)
6:          γ ← COMPUTEOUTPUTS(DS, β, OUT)
7:          for all l ∈ γ do
8:              γ' ← FLIPLITERAL(γ, l)
9:              ω ← FINDMCDIAGNOSIS(DS, β ∧ γ')
10:             if |ω| > q then
11:                 q ← |ω|
12:                 γ ← γ'
13:                 R ← β ∧ γ'
14:             end if
15:         end for
16:         n ← n + 1
17:     until n < N
18:     return R
19: end function
```

Alg. 1 performs N independent attempts (restarts), each one starting from a random observation vector that corresponds to nominal health. This random starting point is computed as follows. First, the RANDOMINPUTS function assigns to each variable in IN a random value, the resulting term is then assigned to β. These random inputs are then fed to the COMPUTEOUTPUTS subroutine which assigns healthy values to the assumable variables, and computes the values of the variables in the output set OUT. This can be done by using a suitable propagation method like Binary Constraint Propagation (Zabih and McAllester 1988). The result of COMPUTEOUTPUTS is then assigned to γ.

Starting from this initial candidate observation $\beta \wedge \gamma$, Alg. 1 attempts to reduce the cardinality of a minimal-di-

agnosis consistent with an observation vector by "flipping" the values of the output variables. This is achieved by the auxiliary function FLIPLITERAL. At each step, the cardinality of the minimal-cardinality diagnosis is computed by a call to the MBD oracle FINDMCDIAGNOSIS. The observation leading to the highest-cardinality fault is stored and returned as a result of the MFMC computation.

Our MBD oracle must be carefully designed, since computing minimal cardinality diagnoses has a very high worst-case complexity: given arbitrary propositional theories in SD, the complexity of finding the cardinality of a minimal-cardinality diagnosis is $\Sigma_2^{\mathbf{P}}$-hard (Eiter and Gottlob 1995). The complexity decreases by imposing restrictions on the class of admissible system models, e.g., models with ignorance of abnormal behavior (de Kleer, Mackworth, and Reiter 1992), Horn theories, etc. For improving the speed of MBD in the average case, the literature has discussed a number of learning (Williams and Ragno 2007) or approximation (Feldman, Provan, and van Gemund 2007) techniques. Although our MFMC algorithm is transparent to the choice of the minimal-diagnosis oracle, the choice can be optimized when additional information on the specific properties of the system descriptions is available. In our implementation we use SAFARI (Feldman, Provan, and van Gemund 2007) as a function for computing the minimal-diagnosis. SAFARI is a stochastic diagnostic solver which returns minimal diagnoses as an approximation to minimal-cardinality diagnoses but, as we will see later on, the incompleteness is compensated by the superior performance of this method.

We now illustrate the workings of the greedy algorithm on the Boolean subtractor circuit from our running example. We will consider only one run ($N = 1$). The RANDOMINPUTS function can return, for example, an input vector $\beta = \neg x \wedge y \wedge \neg p$. After assuming the "all-healthy" assignment $\omega_6 = h_1 \wedge h_2 \wedge \cdots \wedge h_7$, the subroutine COMPUTEOUTPUTS computes the values of the output variables as $\gamma = d \wedge b$. Our greedy MFMC algorithm first changes the literal b in γ to $\neg b$. The inputs β and the modified γ makes an observation $\alpha_4 = \neg x \wedge y \wedge \neg p \wedge \neg b \wedge d$. The FINDMCDIAGNOSIS function, then, computes that $MinCard(\mathrm{SD} \wedge \alpha_4) = 1$. "Flipping" the sign of the second output variable d in γ leads to an observation $\alpha_5 = \neg x \wedge y \wedge \neg p \wedge \neg b \wedge \neg d$. Diagnosing $\mathrm{SD} \wedge \alpha_5$ results in $MinCard(\mathrm{SD} \wedge \alpha_5) = 2$. In bigger circuits, of course, "flipping" the second variable does not necessarily increase the cardinality of the minimal-cardinality diagnosis. Hence we need multiple attempts, caching the best observation computed so far. At this point there are no more output variables to "flip", hence the run returns α_5 leading to $MFMC(\mathrm{SD}) = 2$.

The number of minimal-cardinality diagnoses MIRANDA computes is determined by the number of restarts N and the number of output variables $|\mathrm{OUT}|$ in a system DS (recall that MIRANDA "flips" only output variables). The outermost loop of Alg. 1 performs N iterations, where in each iteration exactly $|OUT|$ literals are "flipped"; hence, the worst-case complexity is $O(N\,|\mathrm{OUT}|\,\Theta)$, where Θ is the computational complexity of a single minimal-cardinality diagnosis. Every time the sign of a literal is changed, MIRANDA computes a minimal-cardinality diagnosis, which gives us the stated complexity. In particular, with an incomplete diagnostic oracle like SAFARI (Feldman, Provan, and van Gemund 2007) and an incomplete BCP method for consistency checking in the diagnostic procedure, the complexity of MIRANDA becomes $O(N\,|\mathrm{OUT}|\,|\mathrm{COMPS}|\,C)$, where C is the number of clauses in the CNF representation of SD (Zhang and Stickel 2000). This makes our algorithm applicable to larger models. Component-abstraction approaches, e.g., (Siddiqi and Huang 2007), can also further increase the size of models that can be tackled.

Experimental Results

This section discusses some results from an implementation of the MFMC algorithm described above.

Implementation Notes and Test Set Description

We have implemented MIRANDA in approximately 1 000 lines of C code (excluding the MBD oracle code) and it is a part of the LYDIA package. The implementation can be downloaded from `www.fdir.org`.

Traditionally, MBD algorithms have been tested on diagnostic models of digital circuits like the ones included in the ISCAS85 benchmark suite (Brglez and Fujiwara 1985). As models derived from the ISCAS85 circuits are computationally intensive (from a diagnostic perspective), we have also considered four medium-sized circuits from the 74XXX family (Hansen, Yalcin, and Hayes 1999).

Name	Description	|IN|	|OUT|	H	V	C
74182	4-bit CLA	9	5	19	47	75
74L85	4-bit comparator	11	3	33	77	118
74283	4-bit adder	9	5	36	81	122
74181	4-bit ALU	14	8	65	144	228
c432	27-channel int.	36	7	160	356	514
c499	32-bit SEC	41	32	202	445	714
c880	8-bit ALU	60	26	383	826	1 112
c1355	32-bit SEC	41	32	546	1 133	1 610
c1908	16-bit SEC/DEC	33	25	880	1 793	2 378
c2670	12-bit ALU	233	140	1 193	2 695	3 269
c3540	8-bit ALU	50	22	1 669	3 388	4 608
c5315	9-bit ALU	178	123	2 307	4 792	6 693
c6288	32-bit multiplier	32	32	2 416	4 864	7 216
c7552	32-bit adder	207	108	3 512	7 232	9 656

Table 1: An overview of the 74XXX/ISCAS85 circuits (H is the number of assumable variables, V denotes the total number of variables and C is the number of clauses)

The original 74XXX/ISCAS85 circuits (cf. Table 1 for an overview) have been translated from the Netlist format to a representation suitable for MIRANDA. Although our method is not restricted to a certain class of models, for the experimental section in this paper we have generated weak-fault models (i.e., models with only normal behavior defined) for each of the 14 circuits. The construction of the weak-fault models is the same as in our running example. In general, weak-fault models expose higher MFMC values than models of circuits where gates are allowed to be "stuck-at".

All time measurements in this paper are performed on a host with 1.86 GHz Pentium M CPU and 2 Gb of RAM.

Computing MFMC Numbers and Vectors

Even after supplying MIRANDA with a state-of-the-art complete diagnostic solver (Feldman and van Gemund 2006), the only circuits amenable to exhaustively enumerating all possible observation vectors were the ones from the 74XXX family. The exact cardinalities of the minimal-cardinality diagnoses of 74182, 74L85, 74283, and 74181 are 5, 3, 5, and 7, respectively.

Instead of configuring MIRANDA with a fixed number of restarts N, in our first experiment we show the number of restarts necessary for computing optimal MFMC values for the small 74XXX circuits. For this experiment MIRANDA was configured with the same complete diagnostic procedure which was used for the earlier, exhaustive experiment. Our implementation of MIRANDA reached the optimal MFMC values after performing 1.1, 3.4, 207.7, and 174.3 restarts for the 74182, 74L85, 74283, and 74181 circuits, respectively (the numbers are averages over 10 runs). The large value of N for the 74283 circuit arises because it has 2 MFMC observation vectors only. Similarly, the 74181 circuit has 456 observations, leading to an MFMC diagnosis of cardinality 7 from a total of 2^{22} observations.

The running time for finding the optimal 74XXX MFMC values (averaged over 10 runs) varied from 0.01 s for the 74182 circuit to 34.2 min for 74181. The long running time for reaching the MFMC of 74181 model comes from the poor performance of the complete diagnostic procedure we have used (despite the fact that we have employed a state-of-the-art solver). This is not surprising, considering that the computational cost of finding a k-minimal-cardinality diagnosis increases with k.

	$N=1$		$N=256$	
Name	Time [s]	MFMC	MFMC	MFMC_e
74182	0.005	5	5	4.72
74L85	0.007	3	3	2.65
74283	0.011	3	5	4.11
74181	0.038	6	7	6.28
c432	0.135	3	8	5.59
c499	0.944	14	22	22.07
c880	2.458	16	26	17.97
c1355	5.069	9	21	22.07
c1908	9.622	10	21	17.24
c2670	97.332	15	32	37.17
c3540	30.061	19	21	16.87
c5315	315.475	41	55	47.12
c6288	84.069	6	12	16.83
c7552	594.304	22	42	46.19

Table 2: MFMC of the benchmark circuits and total number of tests for multiple-fault diagnosis

To overcome the complexity of using a complete diagnostic procedure, in the rest of our experiments, we have used the incomplete SAFARI algorithm, which is virtually insensitive to k (Feldman, Provan, and van Gemund 2007). The stochastic MBD oracle dramatically increases the performance of MIRANDA, at the price of overestimating the cardinality of a minimal-cardinality diagnosis. While SAFARI returns minimal diagnoses, they are not necessarily minimal-cardinality diagnoses. To some extent the optimistic MFMC values from SAFARI compensate for the pessimistic effect of the limited number of MIRANDA retries, but still cause MIRANDA to produce optimistic MFMC values for the ISCAS85 circuits. A procedure to estimate the actual MFMC values is presented in the next section.

Table 2 shows the MFMC data and run times using MIRANDA and SAFARI. The second and third columns measure the time for executing one run ($N = 1$) and the MFMC value reached during this run, respectively. It is visible that, even with one random climb, the MFMC values are in the worst case within 50% of the best MFMC values we have found. These best MFMC values, shown in the fourth column of Table 2, are computed given 256 restarts. The large number of restarts was necessary for creating a model-based diagnosis benchmark (which is not discussed in this paper). Note that, for the 74XXX models, the MFMC values computed with MIRANDA and SAFARI are the same as the global optima when $N = 256$ and within 60% of the global optima when $N = 1$. The rightmost column of Table 2 is a lower bound of the optimal MFMC value, computed by using an alternative method which we will discuss in the next section.

MFMC Error Bounds for Large Circuits

This section describes an alternative method for estimating the MFMC of a circuit, which overcomes the imprecision of our MFMC vectors. MFMC imprecision arises due to two reasons: the stochastic nature of MIRANDA, and the fact that SAFARI returns approximations to minimal-cardinality diagnoses. Although the method described below does not find the actual MFMC vectors, it can be very precise depending on the circuit topology.

Given a system DS, we denote as $g(\text{DS})$ the pdf of the minimal-cardinalities of the diagnoses of all observations in DS. From G we can compute the MFMC value and the number of MFMC observation vectors in DS. In what follows we will see that a normal distribution can be used as an approximation to G for a large class of DS.

To describe our error bounds, we focus on the partitioned observation vector $\alpha = \text{IN} \cup \text{OUT}$. Given an observation α leading to a k-fault minimal diagnosis, we associate a nominal-diagnosis observation α_n, which may differ from α only in the OUT sub-vector. The number of OUT-values in which α and α_n differ is called the *distance* of α, $D(\text{SD}, \alpha)$. If $n = |\text{OUT}|$ is the number of output variables in SD, then starting from any nominal observation α_n, there are ${}_nC_k$ ways to select a distance-k vector α, each of which corresponds to a diagnosis. In the case where each such diagnosis is a minimum cardinality diagnosis, $g(\text{SD})$ is binomially-distributed. This is true given some assumptions on the model SD (e.g., SD is a weak-fault model of a deterministic Boolean circuit).

Although the above model is an approximation, it provides useful bounds on MFMC errors. For the 74XXX and ISCAS85 benchmarks, the fraction of "m-flips" resulting in minimal-cardinality diagnoses of cardinality smaller than m is relatively small and does not vary significantly for different m.

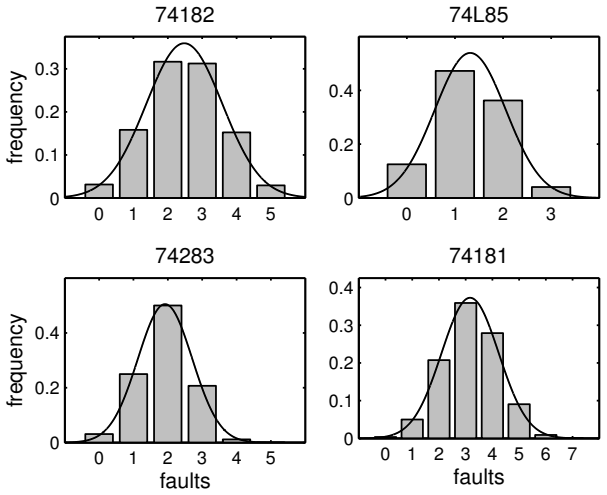

Figure 2: 74XXX minimal-cardinalities pdf

Figure 2 shows a histogram of the true minimal-diagnosis cardinalities for the four 74XXX circuits for which we have exhaustively determined $g(\mathrm{DS})$, fitted by a normal distribution $N(\sigma, \mu)$, denoted $f(x)$ (x is the minimal-cardinality). From $f(0)$ and $f(1)$ it is possible to compute unique values for μ and σ (in practice, we use a numeric method for doing that). Knowing f, the MFMC estimate of the model approximated by f is given[2] by $f^{-1}(f(0))$ for which $f^{-1}(f(0)) \neq 0$.

It is possible to determine $f(0)$ analytically. For any circuit realizing a deterministic Boolean function with disjoint inputs and outputs, i.e., $\mathrm{OBS} = \mathrm{IN} \cap \mathrm{OUT}$ and $\mathrm{IN} \cap \mathrm{OUT} = \emptyset$, it holds that $|\{\alpha : D(\alpha) = 0\}| = 2^{|\mathrm{IN}|}$. These input values produce exactly $2^{|\mathrm{IN}|}$ different observations, hence $f(0) = 2^{|\mathrm{IN}|}/2^{|\mathrm{OBS}|} = 2^{-|\mathrm{OUT}|}$. Finding $f(1)$ is more difficult, and we estimate it by taking S samples of $D(\mathrm{DS}, \alpha) = 2$, and determining the fraction of single fault diagnoses from them. If this fraction is denoted as z, then $f(1) = z\,|\mathrm{OUT}|\,2^{|\mathrm{IN}|}$.

The MFMC estimates, computed with the method outlined above for $S = 1000$, are shown in the rightmost column of Table 2. We can see that for the 74XXX circuits the approximation is within 18%, and that the MFMC lower bound values computed by combinatorial counting are within 69% of the MFMC values returned by MIRANDA for ISCAS85. This gives us an estimate of the error which comes from using the suboptimal SAFARI algorithm as an MBD oracle.

Conclusion

This paper introduced the problem of computing MFMC observation vectors and suggested a greedy stochastic algorithm for computing such vectors. Our algorithm is very efficient, given a fast subroutine for computing the cardinality of a minimal-cardinality diagnosis. The MFMC of real-world systems is an important property quantifying the diagnosability of a model, as it shows the maximum number of malfunctioning components that can be distinguished observing a set of variables. We have empirically demonstrated MIRANDA on a number of 74XXX/ISCAS85 combinatorial circuits, computing the MFMC of these circuits.

In future work we plan to study the coverage of the MFMC observation vectors, in order to compare MFMC-based methods to classical ATPG methods.

[2] f^{-1} is a multi-valued function which has two values in $f(0)$.

References

Abramovici, M. 1981. A maximal resolution guided-probe testing algorithm. In *Proc. DAC'81*, 189–195.

Brglez, F., and Fujiwara, H. 1985. A neutral netlist of 10 combinational benchmark circuits and a target translator in fortran. In *Proc. ISCAS'85*, 695–698.

Console, L.; Picardi, C.; and Ribaudo, M. 2000. Diagnosis and diagnosability analysis using PEPA. In *Proc. ECAI'00*, 131–135.

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

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

Eiter, T., and Gottlob, G. 1995. The complexity of logic-based abduction. *Journal of the ACM* 42(1):3–42.

Feldman, A., and van Gemund, A. 2006. A two-step hierarchical algorithm for model-based diagnosis. In *Proc. AAAI'06*, 827–833.

Feldman, A.; Provan, G.; and van Gemund, A. 2007. Approximate model-based diagnosis using greedy stochastic search. In *Proc. SARA'07*, 139–154.

Hansen, M.; Yalcin, H.; and Hayes, J. 1999. Unveiling the ISCAS-85 benchmarks: A case study in reverse engineering. *IEEE Design & Test* 16(3):72–80.

Hughes, J. L. A. 1988. Multiple fault detection using single fault test sets. *IEEETCAD* 7(1):100–108.

Kubiak, K., and Fuchs, W. K. 1991. Multiple-fault simulation and coverage of deterministic single-fault test sets. In *Proc. of the IEEE International Test Conference*, 956–962.

Provan, G., and Wang, J. 2007. Automated benchmark model generators for model-based diagnostic inference. In *Proc. IJCAI'07*, 513–518.

Roth, J. P. 1966. Diagnosis of automata failures: A calculus and a method. *IBM Journal of R & D* 10:278–291.

Siddiqi, S., and Huang, J. 2007. Hierarchical diagnosis of multiple faults. *Proc. IJCAI'07* 581–586.

Williams, B., and Ragno, R. 2007. Conflict-directed A* and its role in model-based embedded systems. *Journal of Discrete Applied Mathematics* 155(12):1562–1595.

Zabih, R., and McAllester, D. 1988. A rearrangement search strategy for determining propositional satisfiability. In *Proc. AAAI'88*, 155–160.

Zhang, H., and Stickel, M. 2000. Implementing the davis-putnam method. *JAR* 24(1-2):277–296.