A Framework and Algorithm for Model-Based Active Testing

Alexander Feldman, *Student Member, IEEE,* Gregory Provan, *Member, IEEE,* and
Arjan van Gemund, *Member, IEEE Computer Society*

***Abstract*—Due to model uncertainty and/or limited observability, the multiple candidate diagnoses (or the associated probability mass distribution) computed by a Model-Based Diagnosis (MBD) engine may be unacceptable as the basis for important decision-making. In this paper we present a new algorithmic approach, called FRACTAL (FRamework for ACtive Testing ALgorithms), which, given an initial diagnosis, computes the shortest sequence of additional test vectors that minimizes diagnostic entropy. The approach complements probing and sequential diagnosis (ATPG), applying to systems where only additional tests can be performed by using a subset of the existing system inputs while observing the existing outputs (called "Active Testing"). Our algorithm generates test vectors using a myopic, next-best test vector strategy, using a low-cost approximation of diagnostic information entropy to guide the search. Results on a number of** 74XXX/ISCAS85 **combinational circuits show that diagnostic certainty can be significantly increased, even when only a fraction of inputs are available for active testing.**

***Index Terms*—Artificial Intelligence, Model-Based Diagnosis, Troubleshooting.**

I. Introduction

Model-Based Diagnosis (MBD) [1] is an area of abductive inference that uses a system model, together with observations about system behavior to isolate sets of faulty components (diagnoses) that explain the observed behavior. One of the advantages of MBD over related approaches (e.g., simulation-based) is that MBD can cope with arbitrary degree of uncertainty in the system model and in the observation. In the latter case MBD computes *all* or an approximation to all diagnoses. The number of diagnoses can be large, exponential of the number of components in the worst-case.

This ambiguity (uncertainty) of the diagnostic result poses a typical problem to MBD. Due to modeling uncertainty (e.g., weakness due to ignorance of abnormal behavior or need for robustness) and limited number of observations (sensor-lean systems, limited observation horizons), the failure probability mass is distributed over multiple diagnoses. This high information entropy of the diagnostic result makes it difficult for an operator or a reconfiguration (planning) component to decide with sufficient certainty.

Given a set of plausible diagnoses, in certain situations one can devise additional tests that narrow down the ambiguity (reduces the set of diagnoses). When measurements can be made this is a good way to do that [1]. However, in many circumstances there are no provisions for sensing additional variables (e.g., a satellite that cannot be physically reached). In such cases, the only thing that can be done is to actively control (a subset of) inputs, executing a part of the existing system functionality (e.g., invoking built-in test capabilities, or otherwise), the associated observations being used to further narrow down the diagnostic solution space.

Under no constraints, this would mean applying a test vector on *all* inputs such as in sequential diagnosis (and ATPG) where a sequence of tests is applied to target a fault. In many situations, however, this would too much interfere with the system and its environment. Usually, there is a subset of inputs, called control inputs, that can be manipulated by a diagnostic engine to execute tests. This approach is coined "active testing". Loosely speaking, an active testing problem is: given a system model and an initial observation and diagnosis, to compute the set of input test vectors that will minimize diagnostic ambiguity with the least number of test vectors.

In this paper we present a framework, called FRACTAL (FRamework for ACtive Testing ALgorithms), in which we define active testing and present algorithms to solve the active testing problem. Our contributions are as follows:

- We define the active testing problem and describe various instances of the problem;
- We define diagnostic ambiguity in terms of information entropy and propose a low-cost estimation amenable to active testing;
- We define a stochastic, myopic strategy to solving the active testing problem and outline an algorithm to solve the active testing problem;
- We study the performance of our algorithm on the 74XXX/ISCAS85 combinational benchmark suite.

To the best of our knowledge, this is the first approach to defining and solving the active testing problem, generalizing over sequential diagnosis and ATPG. Furthermore our method is based on MBD which is beneficial in that very few assumptions about the model and the observations are required. Our results show that controlling a small fraction of the inputs can reduce the number of remaining diagnoses at a small diagnostic cost whereas a reduction of entropy would be impossible for a passive approach. Our method is also computationally efficient as it uses a stochastic approach and is relevant to practice as it can be effectively used to disambiguate faults in complex autonomous systems.

This paper is organized as follows. The section that comes

Manuscript received May 17, 2008; revised August 15, 2008. This work has been supported by STW grant DES.7015 and SFI grant 07/SRC/I1170.

A. Feldman and A. van Gemund are with the Delft University of Technology, Faculty of Electrical Engineering, Mathematics and Computer Science.

G. Provan is with the University College Cork, Department of Computer Science.

next introduces some basic MBD notions. Section IV presents the problem of sequential MBD and the important concept of remaining number of diagnoses. Section V introduces a framework for active testing. What follows is a section describing algorithms for active testing. Section VII implements the algorithms and cites some experimental results. Finally we summarize our work and discuss future work.

II. Related Work

The problem of sequential diagnosis has received considerable attention in the literature. Our notion of active testing is related to that of Pattipati et al. [2], [3], except that we compute diagnoses rather than caching all diagnoses in a fault dictionary, we assume all tests have identical costs, and we assume all faults are equally likely, a priori. In addition to that, whereas the test matrix in sequential diagnosis is fixed, we allow part of the inputs to be supplied by the environment in every step of the diagnostic process, which makes our framework more suitable for online fault isolation.

Note that our task is harder than that of [3], since they do diagnosis lookup using a fault dictionary, and still show that the sequential diagnosis task is NP-hard; in our case we compute a new diagnosis after every test. Hence we have an NP-hard sequential problem interleaved with the complexity of diagnostic inference at each step.[1]

The framework proposed by Pattipati et al. has been extended to an AND/OR-tree technique that is optimal [4]. We note that optimal test sequencing is infeasible for the size of problems in which we are interested.

Rish et al. [5], [6] define a similar framework, but cast their models in terms of Bayesian networks. Our notion of entropy is the size of the diagnosis space, whereas Rish et al. use decision-theoretic notions of entropy to guide test selection.

The diagnosis framework that we propose is submodular, in the terms described in [7], i.e., the informativeness of tests exhibits diminishing returns the more tests that we do. In future work we plan to compare our stochastic algorithms to the randomized algorithms that have been developed for submodular functions.

In comparison to all of this work, the main contributions of our paper are:

- A model-based framework for combining multiple-fault and sequential diagnosis and the introduction of reasoning with respect to modifiable/non-modifiable observable variables;
- A characterization of diagnostic entropy in terms of the size of the diagnosis space;
- approximating the size of the diagnosis space in terms of the number of different observations;
- A stochastic algorithm for efficiently estimating the number of different observations and resulting diagnoses.

III. Technical Background

Our discussion starts by adopting the relevant MBD notions [1].

Central to MBD, a *model* of an artifact is represented as a propositional $\mathbf{Wff}$ over a set of variables. We will discern three subsets of these variables: *assumable*, *observable*[2] and *control* variables. This gives us our initial definition:

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 $\mathrm{SD} \not\models \perp$. Furthermore, to avoid handling inconsistencies, we restrict SD to models for which $\mathrm{SD} \wedge \alpha \not\models \perp$ for any (possibly partial) assignment α to the variables in OBS.

A. A Running Example

We will use the Boolean circuit shown in Fig. 1 as a running example for illustrating all notions and the algorithm shown in this paper. The 2-to-4 line demultiplexer consists of four Boolean inverters and four and-gates.

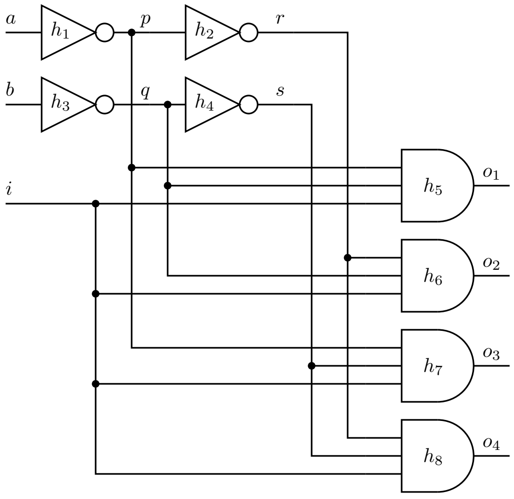

Fig. 1. A demultiplexer circuit

The expression $h \Rightarrow (o \Leftrightarrow \neg i)$ models an inverter, where the variables i, o, and h represent input, output, and health respectively. Similarly, an and-gate is modeled as $h \Rightarrow (o \Leftrightarrow i_1 \wedge i_2 \wedge i_3)$. The above propositional formulae are copied for each gate in Fig. 1 and their variables subscripted and renamed in such a way as to ensure a proper disambiguation and to connect the circuit. The result is the following propositional model:

$$\mathrm{SD} = \begin{cases} [h_1 \Rightarrow (a \Leftrightarrow \neg p)] \wedge [h_2 \Rightarrow (p \Leftrightarrow \neg r)] \\ [h_3 \Rightarrow (b \Leftrightarrow \neg q)] \wedge [h_4 \Rightarrow (q \Leftrightarrow \neg s)] \\ h_5 \Rightarrow (o_1 \Leftrightarrow i \wedge p \wedge q) \\ h_6 \Rightarrow (o_2 \Leftrightarrow i \wedge r \wedge q) \\ h_7 \Rightarrow (o_3 \Leftrightarrow i \wedge p \wedge s) \\ h_8 \Rightarrow (o_4 \Leftrightarrow i \wedge r \wedge s) \end{cases}$$

[1] In our case the complexity of diagnostic inference is Σ_2^p-hard.

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

The assumable variables are $\text{COMPS} = \{h_1, h_2, \ldots, h_8\}$ and the observables are $\text{OBS} = \{a, b, i, o_1, o_2, o_3, o_4\}$. Note the conventional selection of the sign of the "health" variables $h_1, h_2, \ldots, h_n$. Other authors use "ab" for abnormal.

B. Diagnosis

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

Definition 2 (Diagnosis)**.** Given a system DS, an observation α over some variables in OBS, and an assignment ω to all variables in COMPS, ω is a diagnosis iff $\text{SD} \wedge \alpha \wedge \omega \not\models \perp$.

We denote the set of all diagnoses of a model SD and an observation α as $\Omega(\text{SD}, \alpha)$ and the number of all diagnoses as $|\Omega(\text{SD}, \alpha)|$. Continuing our running example, consider an observation vector $\alpha_1 = \neg a \wedge \neg b \wedge i \wedge o_4$. There are a total of 256 possible assignments to all variables in COMPS and $|\Omega(\text{SD}, \alpha_1)| = 200$. Example diagnoses are $\omega_1 = h_1 \wedge h_2 \wedge \ldots \wedge h_7 \wedge \neg h_8$ and $\omega_2 = \neg h_1 \wedge h_2 \wedge h_3 \wedge \neg h_4 \wedge h_5 \wedge h_6 \wedge h_7 \wedge h_8$. We will write sometimes a diagnosis in a set notation, specifying the set of negative literals only. Thus ω_2 would be represented as $D_2 = \{\neg h_1, \neg h_4\}$.

As it is typical for underconstrained models to have many diagnoses (exponential to the number of components in the worst case, as in the above, weak, example model), we will impose (partial) ordering on the diagnoses and will consider only diagnoses which satisfy some minimality criterion.

Definition 3 (Cardinality of a Diagnosis)**.** The cardinality of a diagnosis, denoted as $|\omega|$, is defined as the number of negative literals in ω.

According to Def. 3, we have $|\omega_1| = 1$ and $|\omega_2| = 2$. Next, let us focus on the diagnoses of minimal cardinality.

Definition 4 (Minimal-Cardinality Diagnosis)**.** A diagnosis $\omega^{\leq}$ is defined as Minimal-Cardinality (MC) if no diagnosis $\tilde{\omega}^{\leq}$ exists such that $|\tilde{\omega}^{\leq}| < |\omega^{\leq}|$.

Other authors use different minimality criteria such as subset-minimality diagnoses, probability-minimal diagnoses, kernel diagnoses (in a slightly different diagnostic framework), etc. [8]. Our selection of minimality criterion is such that it does not characterize all diagnoses but is often seen in practice due to the prohibitive cost of computing a characterizing set of diagnoses.

Consider an observation vector $\alpha_2 = \neg a \wedge \neg b \wedge i \wedge \neg o_1 \wedge o_4$. There are 6 MC diagnoses of cardinality 2 consistent with $\text{SD} \wedge \alpha_2$ and counting these MC diagnoses is a common problem in MBD.

Definition 5 (Number of Minimal-Cardinality Diagnoses)**.** The number of MC diagnoses of a system DS given an observation α over some variables in OBS is denoted as $|\Omega^{\leq}(\text{SD}, \alpha)|$, where $\Omega^{\leq}(\text{SD}, \alpha)$ is the set of all MC diagnoses of $\text{SD} \wedge \alpha$.

It is easy to compute the number of MC diagnosis for the circuit in Fig. 1: $|\Omega^{\leq}(\text{SD}, \alpha_1)| = 1$ and $|\Omega^{\leq}(\text{SD}, \alpha_2)| = 6$.

IV. Sequential Diagnosis

Typically, due to uncertainty in the model (e.g., ignorance of abnormal behavior) and in the observation vectors (partial observability), there is more than one MC diagnosis. To reduce this uncertainty and to pinpoint the *exact* cause of failure, diagnosticians often combine a sequence of diagnostic experiments, where, whenever possible, appropriate input vectors are supplied, generating *tests* that optimally reduce $|\Omega|$. If this process of successive application of MBD in time includes dynamic reconfiguration of the system under test, then we call the process *active testing*.

Definition 6 (Diagnostic Sequence)**.** Given a system DS, a diagnostic sequence S is defined as a k-tuple of terms $S = \langle \alpha_1, \alpha_2, \ldots, \alpha_k \rangle$, where α_i $(1 \leq i \leq k)$ is an instantiation of the variables in OBS.

The cost of a diagnostic sequence, denoted as $|S|$, is defined as the number of terms in S (respectively the number of MBD experiments performed by a diagnostician).

An important assumption throughout this paper is that the health of the system under test does not change during the test (i.e., intermittent faults are outside the scope of this study).

Assumption 1 (Non-Intermittence)**.** Given an system DS, an *actual* health state for its components ω_*, and a diagnostic sequence S, we assume that $\omega_* \in \Omega(\text{SD}, \alpha_i)$ for $1 \leq i \leq |S|$.

It is intuitive that for non-intermittent systems, the diagnostician can combine the results from different application of MBD to reduce the diagnostic uncertainty.

Lemma 1. *Given a system* DS*, a health state for its components* ω*, and a diagnostic sequence* S*, it follows that*

$$\omega \in \bigcap_{i=1}^{|S|} \Omega(\text{SD}, \alpha_i)$$

Proof: The above statement follows immediately from the non-intermittence assumption and Def. 2. ■

Obviously, Lemma 1 can be applied only in the cases in which *all* diagnoses of a model and an observation are considered. If we compute minimal-diagnoses in a weak-fault model, for example (cf. [8]), the intersection operator has to be redefined to handle subsumptions. The problem with intersecting diagnostic sets worsens if we consider non-characterizing sets of diagnoses (e.g., MC diagnoses or first n diagnoses). To solve this issue we will provide our own consistency-based intersection operator.

Definition 7 (Consistency-Based Intersection)**.** Given a system description SD, an initial observation α, a (possibly non-characterizing) set of diagnoses D of $\text{SD} \wedge \alpha$, and a posteriori observation α', the intersection of D with the diagnoses of $\text{SD} \wedge \alpha'$, denoted as $\Omega^{\cap}(D, \alpha')$, is defined as the set D' $(D' \subseteq D)$ such that for each $\omega \in D'$ it holds that $\text{SD} \wedge \alpha' \wedge \omega \not\models \perp$.

The intersection operator $\Omega^{\cap}(D, \alpha)$ refines the set of prior diagnoses D, leaving only diagnoses supported by both observations. It is straightforward to generalize the above definition to a diagnostic sequence S.

Definition 8 (Remaining Minimal-Cardinality Diagnoses)**.** Given a diagnostic system DS and a diagnostic sequence S, the set of remaining diagnoses Ω^S is defined as $\Omega^S = \Omega^\cap(\Omega^\cap(\cdots\Omega^\cap(\Omega^\leq(\mathrm{SD},\alpha_1),\alpha_2),\cdots),\alpha_k)$.

It is clear that if we consider the first k terms of a sequence S (forming a subsequence S'), the size of the set of remaining diagnoses $|\Omega^{S'}|$ decreases monotonically when increasing k.

Note that we use $|\Omega^{S'}|$ instead of the more precise diagnostic entropy as defined in [1] and subsequent works. In particular, if all diagnoses of a model and an observation are of minimal-cardinality and the failure probability of each component is the same, then the gain in the diagnostic entropy can be directly computed from $|\Omega^S|$.

V. An Active Testing Framework

Note that in our MBD use of sequential diagnosis, the observation terms are always determined by "nature" [3]. It is often the case, though, that there are inputs (in MBD input and outputs are normally not distinguished and they are both considered as observables) which are not only measurable but also modifiable. We will call these inputs *controls* and we will see that computing values for these control variables can be improve the optimality of the diagnostic process.

A. Optimal Control

Extending the diagnostic system from Def. 1 and separating the controllable from non-controllable observations gives us the following definition:

Definition 9 (Active Testing System)**.** An active testing system ATS is defined as the 4-tuple ATS = $\langle$SD, COMPS, CTL, OBS$\rangle$, where SD is a propositional theory over a set of variables V, COMPS $\subseteq V$, CTL $\subseteq V$, OBS $\subseteq V$, COMPS is the set of assumables, CTL is the set of controls, and OBS is the set of observables.

Furthermore, although this is not strictly necessary, whenever convenient, we will be splitting the set of observables OBS into inputs IN and outputs OUT (OBS = IN $\cup$ OUT, IN $\cap$ OUT = $\emptyset$). Hence, from now on, the observables from the preceding sections will be split into "modifiable" inputs (or controls) CTL, "non-modifiable" inputs IN and outputs OUT. For the assignments to the inputs, outputs, and controls we will conventionally use (subscripted and superscripted when necessary) α, β, and γ, respectively.

Note the distinction between observation terms and control terms. In a typical diagnostic scenario, the observation terms $(\alpha_1, \alpha_2, \ldots, \alpha_k)$ are determined by "nature", while the control terms $(\gamma_1, \gamma_2, \ldots, \gamma_k)$ are set by the diagnostician.

Next, let us consider a diagnostic sequence S whose terms are split into controls and (non-modifiable) inputs ($S = \langle \alpha_1 \wedge \gamma_1, \alpha_2 \wedge \gamma_2, \ldots, \alpha_k \wedge \gamma_k \rangle$). In such a sequence S, a diagnostician would attempt to minimize the set of the remaining diagnoses Ω^S by supplying "optimal" γ_i ($1 \leq i \leq k$) terms. Ideally, there would be exactly one remaining diagnosis ω_* at the end of the sequence. In general, however, there may be more, depending on the model and observability.

Problem 1 (Optimal Control Input)**.** Given a system ATS, and a sequence $S = \langle \alpha_1 \wedge \gamma_1, \alpha_2, \ldots, \alpha_k \rangle$, where α_i ($1 \leq i \leq k$) are OBS assignments and γ_1 is a CTL assignment, compute a minimal sequence of CTL assignments $\gamma_2, \ldots, \gamma_k$, such that $|\Omega^S|$ is minimized.

Problem 1 uses γ_1 because our problem is different from sequential ATPG in the sense that we don't compute tests for specific target diagnosis ω_* (in which case there is no need to have an initial control γ and observation α). In the active testing problem, the situation is different: we target any health state, so initial observation and control are required.

In this paper we will avoid making assumptions on the values of the observable terms $\alpha_1, \alpha_2, \ldots, \alpha_k$. For experimenting with active testing algorithms these can be computed from random inputs and the propagation of the injected fault. There is one special case, however, which is worth distinguishing: $\alpha_1 = \alpha_2 = \cdots = \alpha_k$ (consider, e.g., a system under test which supplies constant observation because it is stationary, paused, pending an abort or reconfiguration, etc.).

Problem 2 (Optimal Control Input for a Persistent Input)**.** Given an active testing system ATS, and a sequence $S = \langle \alpha \wedge \gamma_1, \alpha, \ldots, \alpha \rangle$, where $|S| = k$, α is an OBS assignments and γ_1 is a CTL assignment, compute a minimal sequence of CTL assignments $\gamma_2, \ldots, \gamma_k$, such that $|\Omega^S|$ is minimized.

In practice, a diagnostician does not know what the next observation will be. Fully solving an active testing problem would necessitate the conceptual generation of a tree with all possible observations and associated control assignments in order to choose the sequence that, on average, constitutes the shortest (optimal) path over all possible assignments.

The sequential diagnosis problem studies optimal trees when there is a cost associated with each test [9]. When costs are equal, it can be shown that the optimization problem reduces to a next best control problem (assuming one uses information entropy). In this paper a diagnostician who is given a diagnostic session S and who tries to compute the *next* optimal control assignment would try to minimize the expected number of remaining diagnoses $|\Omega^S|$.

B. Expected Intersection Size

Clearly, $|\Omega^\cap|$ is the goal function to be minimized (apart from k). Next, we will compute the expected number of diagnoses for a set of observable variables M ($M \subseteq$ OBS). Note that the initial observation α and the set of MC diagnoses $D = \Omega^\leq(\mathrm{SD}, \alpha)$ modify the probability density function (pdf) of subsequent outputs[4] (observations), i.e., a subsequent

[3] Note, that in our presentation "sequential diagnosis" is used in the MBD context, which is slightly different from its original presentation, but still compatible. Normally, sequential diagnosis is the art of finding optimal test sequences where typically all inputs are controllable, and where "nature" is only in charge of computing the outputs. In our case, by "nature" we understand the environment (consider the case in which the system description is embedded within a copier that is paused).

[4] In MBD there is no problem not discerning outputs from observables, "assigning values" to outputs, etc. We leave it to the readers' discretion to disambiguate these from the context.

observation α' changes its a priori likelihood. The (non-normalized) a posteriori probability of an observation α', given an MC operator and an initial observation α is:

$$\Pr(\alpha'|\mathrm{SD}, \alpha) = \frac{|\Omega^{\cap}(\Omega^{\leq}(\mathrm{SD}, \alpha), \alpha')|}{|\Omega^{\leq}(\mathrm{SD}, \alpha)|} \tag{1}$$

The above formula comes by quantifying how a given a priori set of diagnoses restricts the possible outputs (i.e., we take as probability the ratio of the number of remaining diagnoses to the number of initial diagnoses). Note that, in practice, there are many α for which $\Pr(\alpha'|\mathrm{SD}, \alpha) = 0$ because a certain fault heavily restricts the possible outputs of a system (i.e., the set of the remaining diagnoses in the nominator is empty).

The expected number of remaining MC diagnoses for a variable set M, given an initial diagnosis α, is then the weighted average of the intersection sizes of all possible instantiations over the variables in M (the weight is the probability of an output):

$$E^{\leq}(\mathrm{SD}, M|\alpha) = \frac{\sum\limits_{\alpha' \in M^*} |\Omega^{\cap}(D, \alpha')| \cdot \Pr(\alpha'|\mathrm{SD}, \alpha)}{\sum\limits_{\alpha' \in M^*} \Pr(\alpha'|\mathrm{SD}, \alpha)} \tag{2}$$

where $D = \Omega^{\leq}(\mathrm{SD}, \alpha)$ and M^* is the set of all possible assignment to the variables in M. Replacing (1) in (2) and simplifying gives us the following definition:

Definition 10 (Expected Minimal-Cardinality Diagnoses Intersection Size)**.** Given a system ATS and an initial observation α, the expected remaining number of MC diagnoses $E^{\leq}(\mathrm{SD}, \mathrm{OBS}|\alpha)$ is defined as:

$$E^{\leq}(\mathrm{SD}, \mathrm{OBS}|\alpha) = \frac{\sum\limits_{\alpha' \in \mathrm{OBS}^*} |\Omega^{\cap}(\Omega^{\leq}(\mathrm{SD}, \alpha), \alpha')|^2}{\sum\limits_{\alpha' \in \mathrm{OBS}^*} |\Omega^{\cap}(\Omega^{\leq}(\mathrm{SD}, \alpha), \alpha')|}$$

where OBS^* is the set of all possible assignment to all variables in OBS.

In what follows we will compute the expected number of remaining MC diagnoses.

VI. An Algorithm for Active Testing

In this section we will consider algorithms for solving the active testing problem. We start with a description of a naïve, exact, table-based method. The memory and time requirements of this exact method are prohibitive, hence the bulk of this section proposes a more efficient, randomized algorithm.

A. Prohibitive Complexity of Exhaustive Search

Consider our running example with an initial observation vector (and control assignment) $\alpha_3 \wedge \gamma_3 = a \wedge b \wedge i \wedge o_1 \wedge \neg o_2 \wedge \neg o_3 \wedge \neg o_4$, where $\gamma_3 = i$ is chosen as the initial control input. The four MC diagnoses of $\mathrm{SD} \wedge \alpha_3 \wedge \gamma_3$ are $D_3 = \{\neg h_1, \neg h_3\}, D_4 = \{\neg h_2, \neg h_5\}, D_5 = \{\neg h_4, \neg h_5\}$, and $D_6 = \{\neg h_5, \neg h_8\}$.

An exhaustive algorithm would compute the expected number of diagnoses for each of the $2^{|\mathrm{CTL}|}$ next possible control assignments. In our running example we have one control variable i and two possible control assignments ($\gamma_5 = i$ and $\gamma_6 = \neg i$). To compute the expected number of diagnoses, for each possible control assignment γ and for each possible observation vector α, we have to count the number of initial diagnoses which are consistent with $\alpha \wedge \gamma$.

Computing the intersection sizes for our running example gives us Table I. Note that, in order to save space, Table I contains rows for those $\alpha \wedge \gamma$ only, for which $\Pr(\alpha \wedge \gamma) \neq 0$, given the initial diagnoses $D_3 - D_6$ (and, as a result, $\Omega^{\cap}(\Omega^{\leq}(\mathrm{SD}, \alpha_3 \wedge \gamma_3), \alpha \wedge \gamma) \neq 0$). It is straightforward to compute the expected number of diagnoses for any control assignment with the help of this marginalization table. In order to do this we have to (1) filter out those lines which are consistent with the control assignment γ and (2) compute the sum and the sum of the squares of the intersection sizes (the rightmost column of Table I). To compute $E(\mathrm{SD}, \mathrm{OBS}|\alpha_3 \wedge \neg i)$, for example, we have to find the sum and the sum of the squares of the intersection sizes of all rows in Table I for which column i is **F**. It can be checked that $E(\mathrm{SD}, \mathrm{OBS}|\alpha_3, \neg i) = 24/16 = 1.5$. Similarly, $E(\mathrm{SD}, \mathrm{OBS}|\alpha_3 \wedge i) = 34/16 = 2.125$. Hence an optimal diagnostician would consider a second measurement with control setting $\gamma = i$.

i	a	b	o_1	o_2	o_3	o_4	Pr	$\lvert\Omega^{\cap}\rvert$
F	**F**	**F**	**F**	**F**	**F**	**F**	0.03125	1
F	**F**	**F**	**T**	**F**	**F**	**F**	0.0625	2
F	**F**	**F**	**T**	**F**	**F**	**T**	0.03125	1
F	**F**	**T**	**F**	**F**	**F**	**F**	0.03125	1
F	**F**	**T**	**T**	**F**	**F**	**F**	0.0625	2
F	**F**	**T**	**T**	**F**	**F**	**T**	0.03125	1
F	**T**	**F**	**F**	**F**	**F**	**F**	0.03125	1
F	**T**	**F**	**T**	**F**	**F**	**F**	0.0625	2
F	**T**	**F**	**T**	**F**	**F**	**T**	0.03125	1
F	**T**	**T**	**F**	**F**	**F**	**F**	0.03125	1
F	**T**	**T**	**T**	**F**	**F**	**F**	0.0625	2
F	**T**	**T**	**T**	**F**	**F**	**T**	0.03125	1
T	**F**	**F**	**F**	**F**	**F**	**T**	0.0625	2
T	**F**	**F**	**F**	**F**	**T**	**F**	0.03125	1
T	**F**	**F**	**F**	**T**	**F**	**F**	0.03125	1
T	**F**	**T**	**F**	**T**	**F**	**F**	0.03125	1
T	**F**	**T**	**T**	**F**	**F**	**F**	0.03125	1
T	**F**	**T**	**T**	**F**	**T**	**T**	0.0625	2
T	**T**	**F**	**F**	**F**	**T**	**F**	0.03125	1
T	**T**	**F**	**T**	**F**	**F**	**F**	0.03125	1
T	**T**	**F**	**T**	**T**	**F**	**T**	0.0625	2
T	**T**	**T**	**T**	**F**	**F**	**F**	0.125	4

TABLE I
Marginalization table for SD and α_3

The obvious problem with the above brute-force approach is that the size of the marginalization table is, in the worst-case, exponential in $|\mathrm{OBS}|$. Although many of the rows in the marginalization table can be skipped as the intersections are empty (there are no consistent prior diagnoses with the respective observation vector and control assignment), the construction of this table is computationally so demanding that we will consider an approximation algorithm (to construct Table 1 for our tiny example, the exhaustive approach had to perform a total of 512 consistency checks).

B. Approximation of the Expectation

Our algorithm for active testing consists of (1) a randomized algorithm for approximating the expected number of remaining diagnoses and (2) a greedy algorithm for searching the space of control assignments. We continue our discussion with approximating the expectation.

The key insight which allows us to build a faster method for computing the expected number of remaining diagnoses is that the prior observation (and respectively a set of MC diagnoses) shifts the probability of the outputs. Hence, an algorithm which samples the possible input assignments (it is safe to assume that inputs are equally likely) and counts the number of *different* observations given the set of prior diagnoses would produce a good approximation. Algorithm 1 uses a couple of auxiliary functions: RANDOMINPUTS assigns random values to all outputs and INFEROUTPUTS computes all outputs from the system model, all inputs and a diagnosis[5]. The computation of the intersection size $|\Omega^\cap(D, \alpha \wedge \beta \wedge \gamma)|$ can be implemented in a straightforward manner. It is enough to count those $\omega \in D$ for which $\mathrm{SD} \wedge \alpha \wedge \beta \wedge \gamma \wedge \omega \not\models \perp$.

Algorithm 1 Approximate expectation

```
function EXPECTATION($(\mathrm{ATS}, \gamma, D)$) returns a real
    inputs: ATS, active testing system
            $\gamma$, term, system configuration
            $D$, set of diagnoses, prior diagnoses
    local variables: $\alpha, \beta, \omega$, terms
                     $s, q$, integers, initially 0
                     $S$, a set of terms, samples
                     $\hat{E}$, a real, expectation
    $S \leftarrow \emptyset$
    repeat
        $\alpha \leftarrow$ RANDOMINPUTS$(\mathrm{SD}, \mathrm{IN})$
        for all $\omega \in D$ do
            $\beta \leftarrow$ INFEROUTPUTS$(\mathrm{SD}, \mathrm{OUT}, \alpha \wedge \gamma, \omega)$
            if $\alpha \wedge \beta \notin S$ then
                $S \leftarrow S \cup \{\alpha \wedge \beta\}$
                $s \leftarrow s + |\Omega^\cap(D, \alpha \wedge \beta \wedge \gamma)|$
                $q \leftarrow q + |\Omega^\cap(D, \alpha \wedge \beta \wedge \gamma)|^2$
                $\hat{E} \leftarrow q/s$
            end if
        end for
    until TERMINATE$(\hat{E})$
    return $\hat{E}$
end function
```

It can be seen that the value of the expected number of diagnoses $\hat{E}$ approaches the exact value E when increasing the number of samples $|S|$. In particular, $\hat{E}$ is the exact number of the expected number of diagnoses when all possible input values are considered (in the latter case Alg. 1 simply builds the marginalization table for a given control assignment γ). Figure 2 shows two examples of $\hat{E}$ approaching E for two bigger models (cf. Sec. VII).

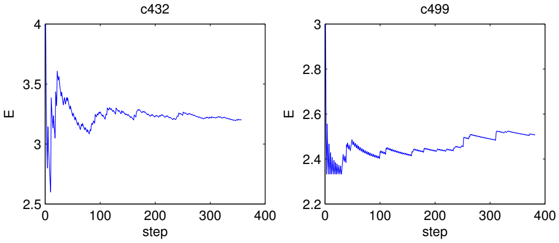

Fig. 2. Example approximations of the expectancy

The algorithm terminates when a suitable termination criterion (checked by the TERMINATE function) is satisfied. In our implementation TERMINATE returns success when the last n iterations (where n is a small constant) leave $\hat{E}$ unchanged.

C. Greedy Control Setting Algorithm

In addition to the approximation for the expectation of the number of diagnoses, we need a faster method for searching the space of possible control assignments (the brute-force approach considers $2^{|\mathrm{CTL}|}$ control assignments). We will assume that the control literals are independent, flip them one at a time, and accept the new control assignment if it decreases the expected number of remaining MC diagnoses. The approach is shown in Alg. 2, which computes a control assignment for a given active testing system and a prior observation. The above algorithm computes a control assignment minimizing the expected intersection size, given an active testing system and an initial observation (and control assignment). The set of initial diagnoses is computed from the initial observation in line 2. In line 5, Alg. 2 "flips" the next literal in the current control assignment. The auxiliary FLIPLITERAL subroutine

Algorithm 2 Optimal next control input

```
1:  function CONTROL($(\mathrm{ATS}, \alpha)$) returns a control term
        inputs: ATS, active testing system
                $\alpha$, term, initial observation
        local variables: $\gamma, \gamma'$, terms, control configurations
                         $E, E'$, reals, expectations
                         $D$, set of terms, diagnoses
                         $l$, literal, control literal
2:      $D \leftarrow \Omega^{\leq}(\mathrm{SD}, \alpha)$
3:      $E \leftarrow$ EXPECTATION$(\mathrm{ATS}, \gamma, D)$
4:      for all $l \in \gamma$ do
5:          $\gamma' \leftarrow$ FLIPLITERAL$(\gamma, l)$
6:          $E' \leftarrow$ EXPECTATION$(\mathrm{ATS}, \gamma', D)$
7:          if $E' < E$ then
8:              $\gamma \leftarrow \gamma'$
9:              $E \leftarrow E'$
10:         end if
11:     end for
12:     return $\gamma$
13: end function
```

[5] This is not always possible in the general case. In our framework, we have a number of assumptions, i.e., a weak-fault model, well-formed circuit, etc. The complexity of INFEROUTPUT varies on the framework and the assumptions.

simply changes the sign of a specified literal in a term. After each "flip" the expected intersection size is computed with a call to EXPECTATION (cf. Alg. 1). If the new expected intersection size is smaller than the current one, then the proposed control assignment is accepted as the current control assignment and the search continues from there.

The advantage of the greedy approach is that the number of computations of expected number of diagnoses is linear of the number of literals in the control assignment. This is done at the price of some optimality (i.e., the effect of combinations of controls is neglected). It is straightforward to turn Alg. 2 into a full heuristic search.

VII. Experimental Results

Next we discuss an implementation of FRACTAL.

A. Experimental Framework for Active Testing

We have implemented a FRACTAL experimental framework. The idea is to (1) inject a random fault and then to (2) simulate a manifestation of this fault. Given this initial manifestation we invoke the active testing algorithm for (3) computing an optimal next control setting. After a control setting is generated, the simulator generates (4) another manifestation of the same fault. This allows FRACTAL to (5) refine the set of diagnoses by intersecting them (cf. Def. 7). Steps 3, 4, and 5 are repeated until some termination criterion is satisfied (e.g., the set of diagnoses remains a singleton[6]).

Algorithm 3 Active testing algorithm

```
1: function ACTIVETEST(ATS) returns a sequence
       inputs: ATS, active testing system
       local variables: $\omega$, term, injected diagnosis
                        $D$, set of terms, diagnoses
                        $\alpha_i, \beta_i, \gamma_i, (1 \leq i \leq k)$, terms
2:     $\omega \leftarrow$ RANDOMFAULT(ATS)
3:     $\alpha_0 \leftarrow$ RANDOMINPUTS(SD, IN)
4:     $\gamma_0 \leftarrow$ RANDOMCONTROLS(SD, CTL)
5:     $\beta_0 \leftarrow$ INFEROUTPUTS(SD, OUT, $\alpha_0 \wedge \gamma_0, \omega$)
6:     $D \leftarrow \Omega^{\leq}(\mathrm{SD}, \alpha_0 \wedge \beta_0 \wedge \gamma_0)$
7:     $k \leftarrow 1$
8:     repeat
9:         $\alpha_k \leftarrow$ RANDOMINPUTS(SD, IN)
10:        $\gamma_k \leftarrow$ CONTROL(SD, $\alpha_{k-1} \wedge \beta_{k-1} \wedge \gamma_{k-1}$)
11:        $\beta_k \leftarrow$ INFEROUTPUTS(SD, OUT, $\alpha_k \wedge \gamma_k, \omega$)
12:        $D \leftarrow \Omega^{\cap}(D, \alpha_k \wedge \beta_k \wedge \gamma_k)$
13:        $k \leftarrow k + 1$
14:    until TERMINATE($D$)
15:    return $\langle \alpha_1 \wedge \beta_1 \wedge \gamma_1, \ldots, \alpha_k \wedge \beta_k \wedge \gamma_k \rangle$
16: end function
```

Algorithm 3 uses the same auxiliary functions RANDOMINPUTS and INFEROUTPUTS as in Sec. VI-B. The subroutine RANDOMCONTROLS is similar to RANDOMINPUTS except that it generates assignments to the variables in CTL instead of the ones in OBS. Similarly, RANDOMFAULT generates a random assignment to the assumable variables in COMPS.

Algorithm 3 maintains a set of remaining diagnoses in D which are iteratively refined in line 12. The algorithm terminates when a suitable termination criterion (checked by the TERMINATE function) is satisfied. In our implementation TERMINATE returns success when there is only one remaining diagnosis in D or when the last n iterations (where n is a small constant) leave the size of D unchanged. Note that depending on the observability of the model (the contents of OBS), it may never happen that D is reduced to a single diagnosis.

B. Implementation Notes and Test Set Description

We have implemented FRACTAL in approximately 1 500 lines of C code (excluding minimal-diagnosis code and consistency checking) and it is a part of the LYDIA package.[7]

Traditionally, MBD algorithms have been tested on diagnostic models of digital circuits like the ones included in the ISCAS85 benchmark suite [10]. 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 [11].

Name	Description	|OBS|	|COMPS|	V	C
74182	4-bit CLA	14	19	47	75
74L85	4-bit comparator	14	33	77	118
74283	4-bit adder	14	36	81	122
74181	4-bit ALU	22	65	144	228
c432	27-channel int.	43	160	356	514
c499	32-bit SEC	73	202	445	714
c880	8-bit ALU	86	383	826	1 112
c1355	32-bit SEC	73	546	1 133	1 610
c1908	16-bit SEC/DEC	58	880	1 793	2 378

TABLE II
An overview of the 74XXX/ISCAS85 circuits (V denotes the total number of variables and C is the number of clauses)

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

C. Performance and Optimality of Active Testing

In this experiment we compare the results of FRACTAL to a setting where all inputs are non-modifiable and the initial observation is repeated at every step of the sequence (cf. Sec. V). Obviously, in the latter case, the initial number of diagnoses can not be reduced any further. The result is shown in the second column of Table III.
The third column of Table III shows the remaining number of diagnoses after $|S|$ steps (column 4 of Table III). Finally, the rightmost column of Table III gives the time (in seconds) for computing the remaining number of diagnoses $\Omega^{\cap}$.

The results show that we can achieve a $25 - 90$ % reduction in the number of diagnoses in $1 - 8$ steps. For these experiments we have set the termination criterion of Alg. 3 to 3

[6] In this case the remaining diagnosis must be the one injected in the beginning of the process (ω_*), which verifies the design and implementation of our algorithms.

[7] LYDIA is downloadable from `http://fdir.org/lydia/`.

Name	$\Omega^{\leq}$	$\Omega^{\cap}$	$\lvert S\rvert$	T [s]
74182	4	2	4	0.6
74L85	8	2	5	2.2
74283	5	3	4	1.5
74181	10	1	2	0.3
c432	10	1	2	20.6
c499	4	4	4	27.6
c880	39	8	4	443.6
c1355	5	4	4	104.1

TABLE III
COMPARISON OF NUMBER OF DIAGNOSES WITH PERSISTENT α TO ACTIVE TESTING ($|\mathrm{CTL}| = |\mathrm{IN}|$)

iterations without changing the number of remaining diagnoses and the precision for Alg. 1 is 0.25. The number of samples for Alg. 1 is 25.

D. *Minimal Expected Intersection Size*

In the experiment that follows we will experiment with computing the expected number of minimal diagnoses with an initial observation only (as opposed to a longer sequence of observations and controls). The result (shown in Table IV) gives an indication on the effect of the control variables on the expected number of remaining diagnoses.

We have seeded our experiments with arbitrary double faults (or single faults for circuits larger than c1355 for faster initial MC computation). The number of initial MC diagnoses is shown in the second column of Table IV. Although the ISCAS85/74XXX benchmark is not designed for active testing, we have "abused" it by changing a fraction $c = |\mathrm{CTL}|/(|\mathrm{CTL}| + |\mathrm{IN}|)$ of the inputs to controls.

Name	$\lvert\Omega^{\leq}\rvert$	$c = 0.25$	$c = 0.5$	$c = 0.75$	$c = 1$
74182	4	3.26	2.36	2.25	2
74L85	8	4.41	4.29	3.63	3
74283	5	3.17	2.25	2.2	2.2
74181	10	6.09	5.28	5.18	3.8
c432	10	3.08	2.93	2.55	2.4
c499	4	4	4	4	4
c880	39	21.68	16.44	15.97	12.23
c1355	5	4.06	3.4	3.4	3.4

TABLE IV
MINIMAL EXPECTED ENTROPY $\hat{E}$ FOR VARYING CONTROLLABILITY ($c = |\mathrm{CTL}|/(|\mathrm{CTL}| + |\mathrm{IN}|)$)

The above experiment shows that a small number of controls ($c = 0.25$) is sufficient for reducing the diagnostic uncertainty.

VIII. CONCLUSION

We have described a framework and an algorithm for active testing, called FRACTAL. The algorithm consists of a sampling-based method for approximating the entropy and a greedy method for searching the next optimal control setting.

We have implemented the algorithm and experimented on a range of combinational benchmarks. Experiments show that controlling a small fraction of the inputs can reduce the diagnostic uncertainty while minimizing the diagnostic cost.

We argue that active testing can be of broad practical significance, as it can reduce diagnostic uncertainty in situations in which MBD alone is not capable of determining exact cause of failure.

As a future work we plan to bound the error of the randomized algorithms and to perform more experiments on additional set of models and observation vectors. We also plan to study the problems complexity and to improve and assess the performance of our implementations.

REFERENCES

[1] J. de Kleer and B. Williams, "Diagnosing multiple faults," *Artificial Intelligence*, vol. 32, no. 1, pp. 97–130, 1987.

[2] K. Pattipati, M. Alexandridis, A. Inc, and M. Burlington, "Application of heuristic search and information theory to sequential fault diagnosis," *IEEE Trans. on SMC*, vol. 20, no. 4, pp. 872–887, 1990.

[3] V. Raghavan, M. Shakeri, K. Pattipati, M. Inc, and M. Natick, "Optimal and near-optimal test sequencing algorithms with realistic test models," *IEEE Trans. on SMC*, vol. 29, no. 1, pp. 11–26, 1999.

[4] O. Kundakcioglu and T. Unluyurt, "Bottom-up construction of minimum-cost and/or trees for sequential fault diagnosis," *IEEE Trans. on SMC*, vol. 37, no. 5, pp. 621–629, 2007.

[5] A. X. Zheng, I. Rish, and A. Beygelzimer, "Efficient test selection in active diagnosis via entropy approximation," in *Proc. UAI'05*, 2005.

[6] M. Brodie, I. Rish, S. Ma, and N. Odin-tsova, "Active probing strategies for problem diagnosis in distributed systems," in *Proc. IJCAI'03*, 2003, pp. 1337–1338.

[7] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *Proc. AAAI'07*, 2007.

[8] J. de Kleer, A. Mackworth, and R. Reiter, "Characterizing diagnoses and systems," *Artificial Intelligence*, vol. 56, no. 2-3, pp. 197–222, 1992.

[9] F. Tu and K. R. Pattipati, "Rollout strategies for sequential fault diagnosis," *IEEE Trans. on SMC*, vol. 33, no. 1, pp. 86–99, 2003.

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

[11] M. Hansen, H. Yalcin, and J. Hayes, "Unveiling the ISCAS-85 benchmarks: A case study in reverse engineering," *IEEE Design & Test*, vol. 16, no. 3, pp. 72–80, 1999.