New Targets for Diagnostic Test Generation

Irith Pomeranz

***Abstract*— A logic diagnosis procedure provides information about the defects that are present in a faulty unit as a set of candidate faults. To obtain smaller, and more accurate, sets of candidate faults, a diagnostic test generation procedure produces a test set that distinguishes fault pairs. This paper observes that large sets of candidate faults are obtained when multiple defects are present in a faulty unit, even if a diagnostic test set is used for logic diagnosis. This points to the possibility that fault pairs do not provide a complete set of targets for diagnostic test generation. The paper analyzes the conditions that cause a large set of candidate faults to be formed under a particular logic diagnosis procedure, and suggests new targets for diagnostic test generation. Experimental results for benchmark circuits demonstrate that a diagnostic test set can be improved by adding diagnostic tests for the new targets.**

***Index Terms*— Diagnostic test generation, logic diagnosis, candidate faults, transition faults.**

I. Introduction

The goal of logic diagnosis is to provide information about the defects that are present in a faulty unit after it had produced a faulty output response to a test set [1]-[16]. The defect information is represented as a set of candidate faults that point to the locations of the defects in the faulty unit. To obtain additional information about the defect locations and types, physical failure analysis may be carried out considering the sites of the candidate faults. The set of candidate faults should be as small as possible to facilitate physical failure analysis.

A two-phase logic diagnosis process is commonly used for reducing the complexity of logic diagnosis [17]. Such a logic diagnosis process applies the same test set to all the units in the first phase. The test set is typically a fault detection test set. Using the same fault detection test set for all the units simplifies the test application process in the first phase. For a faulty unit that produced a faulty output response in the first phase, logic diagnosis is carried out to produce a set of candidate faults. The second phase is applied to faulty units with large sets of candidate faults. In this phase, a diagnostic test set [17]-[36] is computed, and applied to the selected faulty units. The extended output response of the faulty unit is used by the logic diagnosis procedure to produce a smaller, and more accurate, set of candidate faults. Physical failure analysis is applied to faulty units with sufficiently small sets of candidate faults.

Diagnostic test generation is applied to pairs of faults [17]-[36]. A diagnostic test for a pair of faults (f_0, f_1) distinguishes the faults by causing the circuit to produce different output responses for f_0 and f_1. When a pair of faults is distinguished, the logic diagnosis procedure can decide to include only one of the faults in the set of candidate faults. This allows it to produce a smaller set of candidate faults.

With a two-phase logic diagnosis process, diagnostic test generation is carried out only for fault pairs that appear together in large sets of candidate faults [17], [35], [36]. Alternatively, it is possible to apply diagnostic test generation to all the fault pairs that are not distinguished by the fault detection test set [18]-[34]. This allows the diagnostic test set to be generated before logic diagnosis is carried out for any faulty unit. The diagnostic test set is larger in this case, and the computational effort for diagnostic test generation is larger. However, it provides all the diagnostic tests that may be produced based on fault pairs.

This paper observes that, even if a diagnostic test set is used for logic diagnosis, large sets of candidate faults may be obtained when certain multiple defects are present in a faulty unit. A multiple defect consists of two or more single defects that are present in the faulty unit together. Multiple defects of high multiplicities are likely to occur in new technologies. Although the defect multiplicities decrease as the technology matures, the density of devices and the small feature sizes make it common for multiple defects to occur even in mature technologies. When several defects are present in a faulty unit together, their effects may interact. The resulting output response may match that of a fault in a location that is not actually faulty. As a result, the logic diagnosis procedure may include more faults than necessary in the set of candidate faults. The occurrence of large sets of candidate faults in the presence of some multiple defects was demonstrated in [37]-[39]. The experiments in [39] are carried out using a commercial logic diagnosis tool.

The fact that a large set of candidate faults can be obtained even under a diagnostic test set points to the possibility that indistinguished fault pairs do not provide a complete set of targets for diagnostic test generation. The goal of this paper is to analyze the conditions that cause a large set of candidate faults to be formed, and suggest new targets for diagnostic test generation. With the new targets, additional diagnostic tests are generated, and the logic diagnosis procedure yields smaller sets of candidate faults. Consequently, the accuracy of logic diagnosis is improved.

The analysis in this paper is performed with respect to a particular logic diagnosis procedure [15]. The procedure belongs to a class of logic diagnosis procedures that use a set of modeled faults F. By applying fault simulation to F, they assign a score to every fault in F. The faults are ranked based on their scores, and the ranked list of faults is used for defining a set of candidate faults. The defect diagnosis procedures from [1], [2], [7], [9], [11], [13] and [15], as well as commercial tools, belong to this class of procedures. Although the analysis in this paper is performed with the logic diagnosis procedure

Irith Pomeranz is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, U.S.A. (e-mail: pomeranz@ecn.purdue.edu).

This work was supported in part by NSF grant CCF-1714147

from [15], the approach developed based on it is applicable to any logic diagnosis procedure from this class. To support this argument it is important to note that an approach related to the selection of diagnostic tests (different from the one considered in this paper) is shown in [37]-[39] to be applicable to different logic diagnosis procedures.

Diagnostic test generation for the new diagnostic targets starts from an initial test set. For the experiments in this paper, the initial test set is one of three test sets. (1) A diagnostic test set T_{diag} that was generated by targeting the indistinguished fault pairs of a fault detection test set. New targets for diagnostic test generation are computed based on T_{diag}, and new diagnostic tests are generated based on these targets. The use of T_{diag} as an initial test set is important for demonstrating that the new targets are needed even if diagnostic test generation has already been carried out based on fault pairs. (2) Instead of a diagnostic test set, an n-detection test set is also useful for diagnosis (although not as effective as a diagnostic test set). A ten-detection test set T_{10det} is considered in this paper. (3) It is also possible to apply the new targets and diagnostic test generation procedure starting from a fault detection test set, and avoid the need to perform diagnostic test generation based on fault pairs, or n-detection test generation. A fault detection test set T_{fdet} is used as an initial test set to demonstrate this case.

For the experiments in this paper, a simulation framework is used for computing multiple defects with large sets of candidate faults. In a real environment, it is possible to use real defects with large sets of candidate faults. The diagnostic test generation procedure requires knowledge of the defect that is present in the circuit. This information is readily available in a simulation environment. Two sets of experimental results are used for demonstrating that diagnostic tests generated based on certain known defects are useful for other defects as well.

The paper is organized as follows. Section II describes the simulation framework that yields multiple defects with large sets of candidate faults. Section III suggests new targets for diagnostic test generation based on large sets of candidate faults. Section IV describes the diagnostic test generation procedure for the new targets. Experimental results of diagnostic test generation are presented in Section V.

II. Large Sets of Candidate Faults

This section describes a simulation setup where large sets of candidate faults are produced. The setup consists of an iterative procedure that selects multiple defects for simulation, and applies a logic diagnosis procedure to them. Defects that produce large sets of candidate faults are stored in a set D. The notation used for a defect D_i is summarized in Table I.

In the simulation environment used in this paper, a set of faults F is used as a source for defects. A defect D_i of multiplicity $m_i \geq 2$ is obtained by selecting m_i faults from F randomly, and including them in D_i. The only constraint on the selected faults is that D_i should not include two faults on the same line. For example, with transition faults, D_i does not include both the rising and falling transition on the same line. Such faults are detected by different tests, and their effects do not interact to create output responses that are difficult to diagnose.

TABLE I
Notation

symbol	meaning
D_i	defect
m_i	multiplicity, or number of single faults in D_i
C_i	set of candidate faults for D_i
ξ_i	$\|C_i\|/\|D_i\|$, referred to as excess
X_i	$C_i - D_i$
V_i	$C_i \cap D_i$
$\sigma_i(t)$	quality metric for a diagnostic test targeting D_i
α_i	number of diagnostic test generation attempts for D_i
ts	initial test set
ind	index of a defect
mult	multiplicity of the defect
tests	number of tests
att	number of diagnostic test generation attempts
cand	number of candidates
excess	value of the excess
resol	value of the diagnostic resolution
prec	value of the diagnostic precision
diag	number of times logic diagnosis is applied
dtg	number of times diagnostic test generation is applied

TABLE II
Example Set of Defects

i	D_i	C_i	ξ_i
0	266 712	12 71 266 712 763	2.5
1	252 737	123 252 683 716 737	2.5
2	232 271 361	64 120 193 232 255 271 361	2.333
3	155 363 366	93 142 155 251 357 363 366	2.333
4	69 334 551 761	10 14 69 73 264 334 551 758 761	2.25

After D_i is obtained, a logic diagnosis procedure is applied to compute a set of candidate faults C_i. Logic diagnosis is carried out by the procedure from [15]. The set C_i is considered to be large if $|C_i| > 2|D_i|$, i.e., C_i contains more than twice the number of defects present in the faulty unit. In this case, D_i is stored in D. The faults in D_i are removed from F to ensure that they are not used for defining additional defects. This keeps the defects in D disjoint, and ensures that the results are not biased by a particular fault that appears repeatedly as a defect.

The selection of defects is performed with increasing defect multiplicity, $2 \leq \mu \leq 7$. For every value of μ, $10(\mu - 1)$ defects of multiplicity μ are defined, and defects that produce large sets of candidate faults are stored in D. The selection of defects terminates after obtaining 20 defects with large sets of candidate faults.

Suppose that this process yields a set of $N \leq 20$ defects, $D = \{D_0, D_1, ..., D_{N-1}\}$. The corresponding sets of candidate faults are C_0, C_1, ..., C_{N-1}, with $|C_i| > 2|D_i|$ for $0 \leq i < N$. The ratio $|C_i|/|D_i|$ is referred to as the excess, and denoted by ξ_i.

An example of a set D for benchmark circuit $s382$ is shown in Table II. The circuit has 764 transition faults that are included in the set F. The first five defects from D are shown in Table II. For every defect D_i, Table II shows the indices of the faults included in D_i, the indices of the faults included in C_i, and the excess ξ_i.

Other parameters that are used for measuring the effective-

ness of logic diagnosis are the following. The resolution is defined as $|C_i \cap D_i|/|C_i|$. This is the fraction of candidate faults that identify defects from D_i correctly. The precision is defined as $|C_i \cap D_i|/|D_i|$. This is the fraction of defects from D_i that are included in the set of candidate faults. The number of candidate faults $|C_i|$, and the excess ξ_i, are considered in this paper to be more important than the other parameters since a large set of candidate faults prevents physical failure analysis from being carried out.

The logic diagnosis procedure from [15] is effective in diagnosing multiple defects, including ones with high multiplicities. It follows the lines of the procedure from [2], and uses a set F of modeled faults to define sets of candidate faults. Given a defect D_i with a response $R(D_i)$, the procedure computes the output response $R(f_j)$ for every fault $f_j \in F$. By comparing $R(f_j)$ with $R(D_i)$, the procedure computes a score $s(f_j)$ for f_j. The score is equal to the number of bits where $R(f_j)$ is equal to $R(D_i)$. A higher score makes it more likely that f_j is present in the faulty unit. Mismatches between $R(f_j)$ and $R(D_i)$ do not reduce the score since multiple defects result in large numbers of mismatches that do not add diagnostic information when logic diagnosis is based on single faults. Considering every output value separately, the procedure includes in C_i the faults from F with the highest scores that produce the same output value as D_i. This ensures that all the output values of D_i are explained by candidate faults with the highest scores, and contributes to the accuracy of diagnosis.

The set F for the experiments in this paper consists of transition faults. The fault detection test set T_{fdet} and the ten-detection test set T_{10det} are compact test sets for transition faults that consist of broadside and skewed-load tests. The diagnostic test set T_{diag} is produced by the diagnostic test generation procedure described in [29]. This procedure starts from a fault detection test set that consists of broadside and skewed-load tests. The procedure adds broadside and skewed-load tests to distinguish fault pairs that are not distinguished by the fault detection test set. The use of both broadside and skewed-load tests ensures that the test sets detect as many faults as possible, and distinguish as many fault pairs as possible, under the constraints of standard-scan.

III. New Targets for Diagnostic Test generation

This section analyzes the situation where a large set of candidate faults C_i is obtained for a defect D_i, and suggests a new target for diagnostic test generation based on D_i and C_i. The test set used for logic diagnosis is denoted by T, and can be T_{diag}, T_{10det}, or T_{fdet}.

Let $V_i = C_i \cap D_i$ consist of the candidate faults in C_i that are part of the defect D_i. Let $X_i = C_i - D_i$ consist of the candidate faults in C_i that are not part of the defect D_i. These faults are referred to as extra candidate faults. The reason an extra candidate fault $f_j \in X_i$ is included in C_i is that its score is at least as high as those of the faults in V_i. To allow the logic diagnosis procedure to exclude f_j from C_i, the score of f_j needs to be lower than the scores of the faults in V_i. This can be accomplished by adding a new diagnostic test. In general, the diagnostic test needs to have smaller contributions to the scores of the faults in X_i compared with the ones in V_i.

Based on this discussion, a new target for diagnostic test generation is defined based on D_i and C_i as follows. For an arbitrary test t, let the output response of the defect D_i be $R(t, D_i)$. For a fault $f_j \in F$, let the output response to t be $R(t, f_j)$. The number of bits where $R(t, f_j)$ is equal to $R(t, D_i)$ is denoted by $s(t, f_j)$. This is also the contribution of t to the score of f_j if it is added to T. The target for diagnostic test generation is to generate a test t such that $s(t, f_j)$ is as high as possible for a fault $f_j \in V_i$, and as low as possible for a fault $f_j \in X_i$. To capture both targets, the quality of t as a diagnostic test is measured by the parameter

$\sigma_i(t) = \sum\{s(t, f_j) : f_j \in V_i\} - \sum\{s(t, f_j) : f_j \in X_i\}$.

The new target for diagnostic test generation is to generate a test t for which $\sigma_i(t)$ is as high as possible.

This target for diagnostic test generation is different from the target used in [17]-[36], where the goal is to distinguish fault pairs. Even if fault pairs from C_i are distinguished, the scores may not change in a way that will allow the logic diagnosis procedure to reduce the number of extra candidate faults. The new target addresses the scores of individual faults without considering fault pairs.

It is also interesting to note that using fault pairs from C_i creates $|C_i|(|C_i| - 1)/2$ target fault pairs. Only one target is defined for C_i based on $\sigma_i(t)$. Several tests may be generated for C_i by recomputing $\sigma_i(t)$ as T, C_i and X_i change.

IV. New Diagnostic Test Generation Procedure

This section describes the generation of diagnostic tests for the new diagnostic targets defined in Section III. The goal is to complement a given test set, T. The new diagnostic test generation procedure is illustrated by Figure 1.

A. Target Defects

The procedure from Figure 1 considers the defects produced by the simulation framework from Section II one by one.

For a constant A, the procedure performs diagnostic test generation for a defect $D_i \in D$ up to A times, or until $\xi_i = 1$ is obtained. Every time an attempt is made to generate a new diagnostic test for D_i, the procedure increments a variable α_i by one. After α_i reaches A, D_i is not targeted again. The constant $A = 30$ is used in this paper. It is divided as follows.

An attempt to generate a diagnostic test t for D_i starts by initializing t to be equal to one of the tests from T. The test used for initialization of t is selected randomly, but with the following constraints.

A test that detects D_i is more likely to yield a diagnostic test for D_i. Therefore, the constant A is divided into A_1 and A_0 such that $A_1 > A_0$. For initialization of t in different attempts, the procedure uses up to A_1 tests that detect D_i, and up to A_0 tests that do not detect D_i. The constants used for the experiments are $A_1 = 20$ and $A_0 = 10$.

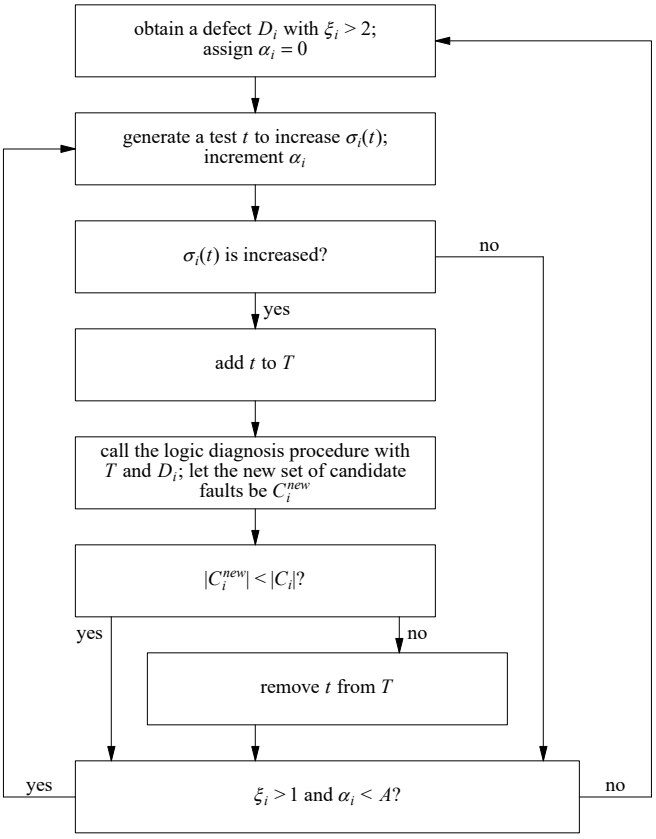

Fig. 1. Diagnostic test generation procedure

B. Generating a New Diagnostic Test

To generate a new diagnostic test t for D_i, the procedure described in this section initializes t such that it is equal to one of the tests in T. It then attempts to increase $\sigma_i(t)$ by modifying t using a simulation-based process.

As an alternative to this process, a deterministic test generation procedure can be modified to target a high value of $\sigma_i(t)$ by adjusting the output values obtained under the faults in V_i and X_i. A simulation-based procedure is preferred in this paper because of its worst-case polynomial time complexity.

After initialization of t, the procedure computes $\sigma_i(t)$. It then attempts to increase $\sigma_i(t)$ by complementing bits of t one at a time in a random order. Let the test t^b be obtained by complementing bit b. The procedure computes $\sigma_i(t^b)$. If $\sigma_i(t^b) \geq \sigma_i(t)$, the procedure accepts the complementation of bit b. In this case it assigns $t = t^b$ and $\sigma_i(t) = \sigma_i(t^b)$. Otherwise, it discards t^b.

The procedure considers every bit of t three times before it accepts t as a new diagnostic test.

The computational complexity of generating a test is determined as follows. Every time the test is modified, the procedure performs fault simulation of the faults in C_i in order to recompute $\sigma_i(t)$. This is typically a small subset of all the possible faults in F. For a circuit with N_{PI} primary inputs and N_{SV} state variables, the number of bits in a test is $O(N_{PI} + N_{SV})$. This is also the number of modified tests that the procedure computes.

C. Evaluating a New Diagnostic Test

Diagnostic test generation is based only on the metric $\sigma_i(t)$. Consequently, it is not guaranteed that adding t to T will reduce the number of candidate faults for D_i. Therefore, the procedure needs to evaluate t and decide whether or not to include it in T. This proceeds as follows.

After a new test t is generated for D_i, the procedure adds t to T temporarily. It then calls the logic diagnosis procedure with T and D_i to compute a new set of candidate faults, C_i^{new}. For t to be considered effective, it is required that the number of candidate faults would be reduced, or $|C_i^{new}| < |C_i|$. If this condition is not satisfied, the procedure removes t from T. Otherwise, t remains in T permanently.

V. Experimental Results

The diagnostic test generation procedure from Figure 1 is applied to benchmark circuits. Table headers for the tables included in this section are summarized in Table I and explained below.

A. Diagnostic Test Generation

The results using the diagnostic test set T_{diag} as an initial test set are shown in Tables III and IV. The number of diagnostic tests that the procedure from Figure 1 adds to the initial test set is denoted by Δ_T. The circuits in Tables III and IV are arranged by increasing value of Δ_T.

The results using the ten-detection test set T_{10det} as an initial test set are shown in Table V. Circuits from Table III are considered, for which the lowest numbers of tests are added to the diagnostic test set T_{diag}.

The results using the fault detection test set T_{fdet} as an initial test set are shown in Tables VI-IX.

The procedure from Figure 1 is run to completion for the circuits in Tables III-VIII. Accordingly, Tables III-VIII describe every defect with a large set of candidate faults for which diagnostic test generation reduces the excess. In addition, it describes the last defect with a large set of candidate faults that the framework from Section II produces, whether or not new diagnostic tests are produced for it. A single row for a defect indicates that no new diagnostic tests are produced.

As the procedure generates more diagnostic tests, it becomes more difficult to find defects with large sets of candidate faults. Only the first such defects are reported for the circuits in Table IX.

It is important to describe the defects with large sets of candidate faults individually since averages hide the differences between them as well as the improvements achieved for them.

For every defect, the first (second) row shows information obtained before (after) diagnostic test generation is carried out for the defect.

After the circuit name, column ts shows whether T_{diag} (1), T_{10det} (2), or T_{fdet} (0) is used as an initial test set. Column ind shows the index i of a defect D_i. Column $mult$ shows its multiplicity m_i. Column $tests$ shows the number of tests in the test set. Column att shows the number of attempts α_i to generate a diagnostic test. The number of attempts starts from zero for every defect.

TABLE III
INITIAL DIAGNOSTIC TEST SET, $\Delta_T < 10$

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec	diag	dtg
b07	1	2	6	136	0	14	2.333	0.429	1.000	207	60
i2c	1	4	7	253	0	15	2.143	0.467	1.000	421	120
sasc	1	1	7	135	0	28	4.000	0.250	1.000	221	30
b04	1	1	7	139	0	15	2.143	0.400	0.857	246	30
b04	1	1	7	140	1	9	1.286	0.667	0.857	247	31
simple_spi	1	0	6	203	0	31	5.167	0.194	1.000	135	0
simple_spi	1	0	6	204	6	30	5.000	0.200	1.000	146	6
pci_spoci_ctrl	1	13	5	203	0	14	2.800	0.357	1.000	565	390
pci_spoci_ctrl	1	13	5	204	28	13	2.600	0.385	1.000	604	418
pci_spoci_ctrl	1	14	5	204	0	21	4.200	0.238	1.000	609	420
pci_spoci_ctrl	1	14	5	205	1	18	3.600	0.278	1.000	610	421
pci_spoci_ctrl	1	19	5	205	0	15	3.000	0.267	0.800	869	570
s526	1	1	3	123	0	9	3.000	0.333	1.000	46	30
s526	1	1	3	124	27	6	2.000	0.500	1.000	89	57
s526	1	3	4	124	0	11	2.750	0.364	1.000	168	90
s526	1	3	4	125	24	10	2.500	0.400	1.000	203	114
s526	1	8	6	125	0	15	2.500	0.400	1.000	438	235
s526	1	8	6	126	1	13	2.167	0.462	1.000	439	236
s526	1	11	7	126	0	18	2.571	0.389	1.000	641	325
s953	1	0	5	246	0	11	2.200	0.364	0.800	87	0
s953	1	0	5	248	8	8	1.600	0.500	0.800	93	8
s953	1	1	7	248	0	17	2.429	0.412	1.000	212	30
s953	1	1	7	249	24	16	2.286	0.438	1.000	257	54
usb_phy	1	2	4	129	0	11	2.750	0.364	1.000	137	60
usb_phy	1	2	4	131	13	9	2.250	0.444	1.000	161	73
usb_phy	1	4	7	131	0	16	2.286	0.438	1.000	422	120
usb_phy	1	4	7	132	16	14	2.000	0.500	1.000	451	136
b09	1	0	3	68	0	8	2.667	0.250	0.667	28	0
b09	1	0	3	69	4	7	2.333	0.286	0.667	35	4
b09	1	0	3	70	5	6	2.000	0.333	0.667	36	5
b09	1	1	5	70	0	11	2.200	0.455	1.000	121	30
b09	1	1	5	71	1	10	2.000	0.400	0.800	122	31
b09	1	3	7	71	0	19	2.714	0.316	0.857	360	90
b09	1	3	7	72	9	18	2.571	0.333	0.857	375	99
s382	1	1	3	65	0	8	2.667	0.375	1.000	59	30
s382	1	1	3	67	3	5	1.667	0.600	1.000	63	33
s382	1	6	6	67	0	15	2.500	0.400	1.000	378	180
s382	1	6	6	68	11	12	2.000	0.417	0.833	399	191
s382	1	7	6	68	0	14	2.333	0.429	1.000	449	210
s382	1	7	6	70	8	12	2.000	0.500	1.000	463	218
s382	1	15	7	70	0	20	2.857	0.350	1.000	945	450
s382	1	15	7	71	17	18	2.571	0.389	1.000	978	467
s382	1	16	7	71	0	15	2.143	0.467	1.000	1002	480
b08	1	2	4	128	0	9	2.250	0.444	1.000	53	36
b08	1	2	4	129	14	4	1.000	0.750	0.750	62	50
b08	1	3	4	129	0	12	3.000	0.250	0.750	71	50
b08	1	3	4	131	14	3	0.750	1.000	0.750	87	64
b08	1	4	4	131	0	12	3.000	0.333	1.000	92	64
b08	1	4	4	133	26	6	1.500	0.667	1.000	136	90
b08	1	7	5	133	0	11	2.200	0.364	0.800	266	154
b08	1	7	5	134	6	10	2.000	0.400	0.800	277	160
b08	1	11	7	134	0	16	2.286	0.375	0.857	578	274
b08	1	11	7	135	22	15	2.143	0.400	0.857	621	296
b08	1	12	7	135	0	19	2.714	0.316	0.857	649	304
b03	1	0	4	71	0	10	2.500	0.400	1.000	35	0
b03	1	0	4	73	29	8	2.000	0.500	1.000	83	29
b03	1	1	5	73	0	11	2.200	0.455	1.000	127	30
b03	1	1	5	75	16	9	1.800	0.556	1.000	155	46
b03	1	2	7	75	0	18	2.571	0.333	0.857	243	60
b03	1	2	7	76	6	17	2.429	0.353	0.857	254	66
b03	1	3	7	76	0	26	3.714	0.192	0.714	352	90
b03	1	3	7	79	11	21	3.000	0.238	0.714	371	101

TABLE IV
INITIAL DIAGNOSTIC TEST SET, $\Delta_T \geq 10$

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec	diag	dtg
steppermotordrive	1	3	5	87	0	13	2.600	0.385	1.000	181	90
steppermotordrive	1	3	5	89	6	11	2.200	0.455	1.000	191	96
steppermotordrive	1	4	5	89	0	15	3.000	0.333	1.000	239	120
steppermotordrive	1	4	5	91	18	10	2.000	0.500	1.000	271	138
steppermotordrive	1	6	6	91	0	18	3.000	0.278	0.833	386	180
steppermotordrive	1	6	6	93	26	14	2.333	0.357	0.833	436	206
steppermotordrive	1	8	6	93	0	17	2.833	0.235	0.667	524	240
steppermotordrive	1	8	6	94	10	16	2.667	0.188	0.500	541	250
steppermotordrive	1	9	6	94	0	14	2.333	0.429	1.000	581	270
steppermotordrive	1	9	6	96	8	11	1.833	0.545	1.000	595	278
steppermotordrive	1	14	7	96	0	16	2.286	0.312	0.714	907	420
steppermotordrive	1	14	7	97	30	14	2.000	0.357	0.714	956	450
steppermotordrive	1	15	7	97	0	16	2.286	0.375	0.857	961	450
steppermotordrive	1	15	7	99	4	12	1.714	0.583	1.000	967	454
b05	1	1	4	190	0	9	2.250	0.444	1.000	115	30
b05	1	1	4	191	2	8	2.000	0.500	1.000	118	32
b05	1	2	5	191	0	11	2.200	0.455	1.000	181	60
b05	1	2	5	192	5	9	1.800	0.556	1.000	188	65
b05	1	3	5	192	0	20	4.000	0.200	0.800	208	90
b05	1	3	5	195	30	11	2.200	0.455	1.000	263	120
b05	1	7	6	195	0	13	2.167	0.462	1.000	481	210
b05	1	7	6	197	16	11	1.833	0.545	1.000	511	226
b05	1	11	7	197	0	20	2.857	0.300	0.857	759	330
b05	1	11	7	199	3	17	2.429	0.353	0.857	763	333
b05	1	13	7	199	0	15	2.143	0.467	1.000	885	390
b05	1	13	7	200	4	14	2.000	0.500	1.000	892	394
b05	1	14	7	200	0	19	2.714	0.368	1.000	947	420
b05	1	14	7	201	19	18	2.571	0.389	1.000	982	439
b05	1	16	7	201	0	20	2.857	0.350	1.000	1088	480
b05	1	16	7	203	19	18	2.571	0.389	1.000	1124	499
b11	1	0	3	187	0	8	2.667	0.375	1.000	12	0
b11	1	0	3	189	4	5	1.667	0.600	1.000	18	4
b11	1	3	5	189	0	13	2.600	0.385	1.000	187	88
b11	1	3	5	190	23	12	2.400	0.417	1.000	228	111
b11	1	4	5	190	0	12	2.400	0.333	0.800	258	118
b11	1	4	5	192	5	9	1.800	0.444	0.800	266	123
b11	1	6	6	192	0	14	2.333	0.429	1.000	427	178
b11	1	6	6	195	18	10	1.667	0.600	1.000	458	196
b11	1	7	7	195	0	18	2.571	0.389	1.000	511	208
b11	1	7	7	197	7	16	2.286	0.438	1.000	521	215
b11	1	9	7	197	0	16	2.286	0.438	1.000	641	268
b11	1	9	7	199	3	14	2.000	0.429	0.857	645	271
b11	1	10	7	199	0	22	3.143	0.318	1.000	689	298
b11	1	10	7	202	22	15	2.143	0.467	1.000	730	320
b11	1	11	7	202	0	17	2.429	0.412	1.000	751	328
b11	1	11	7	203	1	15	2.143	0.467	1.000	752	329
b11	1	12	7	203	0	19	2.714	0.316	0.857	822	358
s1423	1	2	4	169	0	9	2.250	0.444	1.000	44	54
s1423	1	2	4	171	5	6	1.500	0.667	1.000	50	59
s1423	1	3	5	171	0	11	2.200	0.455	1.000	127	84
s1423	1	3	5	173	9	8	1.600	0.625	1.000	143	93
s1423	1	4	6	173	0	18	3.000	0.278	0.833	206	114
s1423	1	4	6	175	7	16	2.667	0.375	1.000	216	121
s1423	1	5	6	175	0	24	4.000	0.250	1.000	266	144
s1423	1	5	6	183	26	11	1.833	0.545	1.000	308	170
s1423	1	6	6	183	0	13	2.167	0.462	1.000	320	174
s1423	1	7	6	185	10	14	2.333	0.429	1.000	407	214
s1423	1	8	6	185	0	14	2.333	0.429	1.000	449	234
s1423	1	8	6	186	16	13	2.167	0.462	1.000	476	250
s1423	1	9	6	186	0	20	3.333	0.300	1.000	507	264
s1423	1	9	6	189	23	16	2.667	0.312	0.833	550	287
s1423	1	10	7	189	0	17	2.429	0.294	0.714	575	294
s1423	1	10	7	191	5	15	2.143	0.400	0.857	583	299
s1423	1	11	7	191	0	19	2.714	0.368	1.000	633	324
s1423	1	11	7	193	6	11	1.571	0.636	1.000	643	330
s1423	1	12	7	193	0	16	2.286	0.438	1.000	692	354
s1423	1	12	7	196	22	12	1.714	0.583	1.000	719	376
s1423	1	14	7	196	0	16	2.286	0.438	1.000	835	414

Column $cand$ shows the number of candidate faults in C_i. Column $excess$ shows the excess ξ_i. Columns $resol$ and $prec$ show the resolution and precision for D_i, respectively.

Column $diag$ shows the total number of times logic diagnosis is applied to one defect, and column dtg shows the total number of times diagnostic test generation is applied to produce one test. These parameters are cumulative, and measure the computational effort. The runtime for computing a diagnostic test using the procedure from Figure 1 is comparable to the runtime for computing a diagnostic test targeting a fault pair using the procedure from [29]. The number of fault pairs that need to be targeted in [29] is typically in the hundreds or thousands, significantly larger than the numbers

TABLE V
INITIAL TEN-DETECTION TEST SET

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec	diag	dtg
b07	2	3	5	603	0	11	2.200	0.364	0.800	220	90
b07	2	3	5	604	1	10	2.000	0.400	0.800	221	91
b07	2	4	5	604	0	11	2.200	0.455	1.000	278	120
b07	2	4	5	606	15	9	1.800	0.556	1.000	302	135
b07	2	6	5	606	0	18	3.600	0.278	1.000	393	180
b07	2	6	5	607	4	17	3.400	0.294	1.000	400	184
b07	2	8	5	607	0	15	3.000	0.267	0.800	521	240
b07	2	8	5	608	2	14	2.800	0.286	0.800	524	242
b07	2	11	6	608	0	14	2.333	0.429	1.000	739	330
b07	2	11	6	609	8	11	1.833	0.545	1.000	754	338
b07	2	14	6	609	0	13	2.167	0.462	1.000	913	420
b07	2	14	6	610	10	12	2.000	0.500	1.000	932	430
b07	2	19	7	610	0	15	2.143	0.400	0.857	1264	570
b04	2	1	4	413	0	9	2.250	0.444	1.000	93	30
b04	2	1	4	417	20	5	1.250	0.800	1.000	119	50
b04	2	2	4	417	0	13	3.250	0.308	1.000	135	60
b04	2	2	4	418	16	12	3.000	0.333	1.000	166	76
b04	2	6	6	418	0	34	5.667	0.147	0.833	452	180
b04	2	6	6	421	26	30	5.000	0.167	0.833	501	206
b04	2	7	7	421	0	21	3.000	0.333	1.000	559	210
b04	2	7	7	424	12	18	2.571	0.389	1.000	580	222
b04	2	8	7	424	0	15	2.143	0.467	1.000	619	240
b04	2	8	7	426	6	13	1.857	0.538	1.000	627	246
s526	2	9	4	700	0	17	4.250	0.235	1.000	319	270
s526	2	9	4	701	1	16	4.000	0.250	1.000	320	271
s526	2	10	4	701	0	14	3.500	0.214	0.750	384	300
s526	2	10	4	703	15	12	3.000	0.250	0.750	412	315
s526	2	14	5	703	0	12	2.400	0.417	1.000	581	420
s526	2	14	5	704	5	7	1.400	0.714	1.000	586	425
s526	2	15	5	704	0	16	3.200	0.312	1.000	622	450
s526	2	15	5	705	2	15	3.000	0.333	1.000	625	452
s526	2	19	6	705	0	15	2.500	0.400	1.000	870	570
s953	2	4	6	1038	0	13	2.167	0.462	1.000	309	120
s953	2	4	6	1039	27	11	1.833	0.545	1.000	350	147
s953	2	5	6	1039	0	13	2.167	0.462	1.000	363	150
s953	2	5	6	1040	21	12	2.000	0.500	1.000	392	171
s953	2	7	7	1040	0	24	3.429	0.292	1.000	466	210
s953	2	7	7	1042	22	21	3.000	0.286	0.857	508	232
s953	2	8	7	1042	0	23	3.286	0.217	0.714	532	240
s953	2	8	7	1043	9	22	3.143	0.227	0.714	549	249
s953	2	9	7	1043	0	16	2.286	0.375	0.857	594	270
s953	2	9	7	1044	1	12	1.714	0.417	0.714	595	271
s953	2	11	7	1044	0	15	2.143	0.467	1.000	720	330
s953	2	11	7	1045	6	14	2.000	0.500	1.000	729	336
s953	2	12	7	1045	0	16	2.286	0.438	1.000	758	360
b09	2	1	5	296	0	12	2.400	0.417	1.000	121	30
b09	2	1	5	297	17	11	2.200	0.455	1.000	152	47
b09	2	2	6	297	0	13	2.167	0.385	0.833	205	60
b09	2	2	6	298	1	12	2.000	0.417	0.833	206	61
b09	2	3	6	298	0	14	2.333	0.357	0.833	265	90
b09	2	3	6	299	2	13	2.167	0.385	0.833	268	92
b09	2	10	7	299	0	16	2.286	0.312	0.714	746	300
b09	2	10	7	300	2	15	2.143	0.333	0.714	749	302
s382	2	6	5	339	0	11	2.200	0.455	1.000	408	180
s382	2	6	5	340	7	10	2.000	0.500	1.000	421	187
s382	2	16	7	340	0	15	2.143	0.467	1.000	1045	480
b08	2	0	2	554	0	5	2.500	0.400	1.000	9	0
b08	2	0	2	555	1	2	1.000	1.000	1.000	10	1
b08	2	3	3	555	0	7	2.333	0.429	1.000	142	61
b08	2	3	3	557	27	3	1.000	1.000	1.000	184	88
b08	2	5	4	557	0	10	2.500	0.400	1.000	270	118
b08	2	5	4	558	9	9	2.250	0.444	1.000	285	127
b08	2	8	6	558	0	19	3.167	0.316	1.000	476	208
b08	2	8	6	559	1	18	3.000	0.333	1.000	477	209
b08	2	8	6	560	20	17	2.833	0.353	1.000	510	228
b08	2	9	7	560	0	15	2.143	0.467	1.000	566	238
b08	2	9	7	561	4	14	2.000	0.500	1.000	571	242
b08	2	10	7	561	0	16	2.286	0.375	0.857	620	268
b08	2	10	7	562	1	15	2.143	0.400	0.857	621	269
b08	2	13	7	562	0	15	2.143	0.400	0.857	821	358

TABLE VI
INITIAL FAULT DETECTION TEST SET, $s5378$, $s9234$ AND $b14$

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec	diag	dtg
s5378	0	3	2	229	0	13	6.500	0.154	1.000	110	73
s5378	0	3	2	230	3	2	1.000	1.000	1.000	111	76
s5378	0	7	3	230	0	9	3.000	0.333	1.000	262	166
s5378	0	7	3	232	20	6	2.000	0.500	1.000	274	186
s5378	0	8	3	232	0	16	5.333	0.188	1.000	289	196
s5378	0	8	3	235	25	13	4.333	0.231	1.000	306	221
s5378	0	18	4	235	0	12	3.000	0.333	1.000	733	491
s5378	0	18	4	236	10	10	2.500	0.400	1.000	750	501
s5378	0	19	4	236	0	9	2.250	0.444	1.000	787	521
s5378	0	19	4	237	3	7	1.750	0.571	1.000	790	524
s9234	0	10	3	344	0	19	6.333	0.158	1.000	292	291
s9234	0	10	3	345	5	18	6.000	0.167	1.000	297	296
s9234	0	11	3	345	0	10	3.333	0.300	1.000	324	321
s9234	0	11	3	346	18	9	3.000	0.333	1.000	337	339
s9234	0	13	3	346	0	14	4.667	0.214	1.000	385	381
s9234	0	13	3	348	13	8	2.667	0.375	1.000	387	394
s9234	0	14	3	348	0	8	2.667	0.375	1.000	388	398
s9234	0	14	3	349	14	6	2.000	0.500	1.000	395	412
s9234	0	17	3	349	0	10	3.333	0.300	1.000	455	473
s9234	0	17	3	350	2	9	3.000	0.333	1.000	458	475
s9234	0	19	3	350	0	13	4.333	0.231	1.000	508	533
s9234	0	19	3	351	16	7	2.333	0.429	1.000	519	549
b14	0	0	2	183	0	21	10.500	0.095	1.000	2	0
b14	0	0	2	184	1	20	10.000	0.100	1.000	3	1
b14	0	2	3	184	0	7	2.333	0.429	1.000	85	53
b14	0	2	3	185	5	4	1.333	0.500	0.667	90	58
b14	0	3	4	185	0	12	3.000	0.333	1.000	153	83
b14	0	3	4	187	16	7	1.750	0.571	1.000	173	99
b14	0	5	5	187	0	37	7.400	0.135	1.000	277	143
b14	0	5	5	188	29	36	7.200	0.139	1.000	326	172
b14	0	6	5	188	0	111	22.200	0.045	1.000	331	173
b14	0	6	5	192	26	35	7.000	0.114	0.800	357	199
b14	0	9	5	192	0	40	8.000	0.125	1.000	452	263
b14	0	9	5	194	16	38	7.600	0.132	1.000	468	279
b14	0	10	5	194	0	40	8.000	0.125	1.000	486	293
b14	0	10	5	197	23	26	5.200	0.192	1.000	511	316
b14	0	11	5	197	0	11	2.200	0.455	1.000	521	323
b14	0	11	5	198	17	10	2.000	0.500	1.000	530	340
b14	0	12	6	198	0	38	6.333	0.158	1.000	547	353
b14	0	12	6	201	11	9	1.500	0.556	0.833	564	364
b14	0	14	6	201	0	19	3.167	0.263	0.833	677	413
b14	0	14	6	202	26	7	1.167	0.714	0.833	712	439
b14	0	15	6	202	0	13	2.167	0.385	0.833	722	443
b14	0	15	6	203	7	8	1.333	0.500	0.667	735	450
b14	0	16	6	203	0	21	3.500	0.190	0.667	783	473
b14	0	16	6	205	23	18	3.000	0.222	0.667	827	496
b14	0	17	7	205	0	46	6.571	0.152	1.000	859	503
b14	0	17	7	206	3	45	6.429	0.156	1.000	862	506
b14	0	18	7	206	0	16	2.286	0.375	0.857	912	533
b14	0	18	7	208	25	13	1.857	0.462	0.857	958	558
b14	0	19	7	208	0	18	2.571	0.278	0.714	969	563
b14	0	19	7	209	13	17	2.429	0.294	0.714	994	576

of diagnostic test generation attempts in Tables III-IX.

The following points can be seen from Tables III-IX. Even when the initial test set is a diagnostic test set that targets all the indistinguished fault pairs of a fault detection test set, large sets of candidate faults are obtained. Defects with low multiplicities are sufficient for creating large sets of candidate faults. Moreover, the procedure from Figure 1 is able to generate additional diagnostic tests that yield reduced sets of candidates faults.

In addition to reducing the number of candidate faults, the new diagnostic tests reduce the excess, and increase the resolution. The precision typically remains the same. A loss of precision is possible when the number of candidate faults is reduced since the intersection between the defect and the set of candidate faults may decrease together with the number of candidate faults. Nevertheless, there are also cases where the precision increases. The precision is allowed to change in order to support a reduction in the number of candidate faults and the excess, which are considered to be more important as discussed earlier.

The procedure typically finds defects with large excess

TABLE VII
INITIAL FAULT DETECTION TEST SET, *des_area* AND *spi*

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec	diag	dtg
des_area	0	1	3	123	0	8	2.667	0.375	1.000	77	30
des_area	0	1	3	125	4	4	1.333	0.750	1.000	81	34
des_area	0	2	3	125	0	8	2.667	0.375	1.000	98	48
des_area	0	2	3	127	2	5	1.667	0.600	1.000	100	50
des_area	0	3	5	127	0	11	2.200	0.455	1.000	145	62
des_area	0	3	5	129	3	7	1.400	0.714	1.000	149	65
des_area	0	4	5	129	0	12	2.400	0.417	1.000	202	92
des_area	0	4	5	131	2	9	1.800	0.556	1.000	204	94
des_area	0	5	5	131	0	14	2.800	0.357	1.000	276	122
des_area	0	5	5	133	5	12	2.400	0.417	1.000	284	127
des_area	0	7	5	133	0	11	2.200	0.364	0.800	404	182
des_area	0	7	5	135	9	6	1.200	0.667	0.800	408	191
des_area	0	8	6	135	0	14	2.333	0.429	1.000	475	212
des_area	0	8	6	139	11	7	1.167	0.857	1.000	493	223
des_area	0	9	6	139	0	13	2.167	0.462	1.000	529	242
des_area	0	9	6	141	7	9	1.500	0.667	1.000	541	249
des_area	0	10	6	141	0	15	2.500	0.400	1.000	574	272
des_area	0	10	6	145	11	7	1.167	0.857	1.000	588	283
des_area	0	11	6	145	0	13	2.167	0.462	1.000	623	302
des_area	0	11	6	147	15	11	1.833	0.545	1.000	651	317
des_area	0	12	6	147	0	14	2.333	0.429	1.000	662	332
des_area	0	12	6	149	2	10	1.667	0.600	1.000	664	334
des_area	0	13	7	149	0	17	2.429	0.353	0.857	713	359
des_area	0	13	7	151	14	15	2.143	0.333	0.714	739	373
des_area	0	14	7	151	0	19	2.714	0.368	1.000	790	389
des_area	0	14	7	152	28	18	2.571	0.389	1.000	845	417
des_area	0	15	7	152	0	20	2.857	0.350	1.000	852	419
des_area	0	15	7	154	4	17	2.429	0.412	1.000	858	423
spi	0	0	2	513	0	8	4.000	0.250	1.000	3	0
spi	0	0	2	514	1	6	3.000	0.333	1.000	4	1
spi	0	1	2	514	0	7	3.500	0.286	1.000	12	16
spi	0	1	2	516	10	3	1.500	0.667	1.000	14	26
spi	0	2	3	516	0	8	2.667	0.375	1.000	18	29
spi	0	2	3	517	2	5	1.667	0.600	1.000	21	31
spi	0	4	3	517	0	10	3.333	0.300	1.000	77	89
spi	0	4	3	518	7	9	3.000	0.333	1.000	86	96
spi	0	6	4	518	0	9	2.250	0.444	1.000	206	149
spi	0	6	4	519	8	8	2.000	0.500	1.000	221	157
spi	0	7	5	519	0	11	2.200	0.455	1.000	279	179
spi	0	7	5	520	4	9	1.800	0.556	1.000	286	183
spi	0	9	5	520	0	14	2.800	0.357	1.000	351	239
spi	0	9	5	521	5	9	1.800	0.556	1.000	360	244
spi	0	11	5	521	0	13	2.600	0.385	1.000	470	299
spi	0	11	5	522	13	9	1.800	0.556	1.000	489	312
spi	0	12	5	522	0	12	2.400	0.417	1.000	494	329
spi	0	12	5	523	22	10	2.000	0.500	1.000	531	351
spi	0	13	5	523	0	14	2.800	0.357	1.000	546	359
spi	0	13	5	524	10	11	2.200	0.455	1.000	557	369
spi	0	14	6	524	0	13	2.167	0.462	1.000	598	389
spi	0	14	6	525	4	11	1.833	0.545	1.000	605	393
spi	0	14	6	526	8	10	1.667	0.600	1.000	610	397
spi	0	16	6	526	0	19	3.167	0.316	1.000	689	449
spi	0	16	6	531	21	10	1.667	0.500	0.833	714	470
spi	0	17	6	531	0	13	2.167	0.462	1.000	728	479
spi	0	17	6	532	8	10	1.667	0.600	1.000	741	487
spi	0	18	6	532	0	26	4.333	0.192	0.833	778	509
spi	0	18	6	534	22	8	1.333	0.625	0.833	820	531
spi	0	19	6	534	0	13	2.167	0.462	1.000	836	539
spi	0	19	6	535	25	11	1.833	0.545	1.000	875	564

TABLE VIII
INITIAL FAULT DETECTION TEST SET, *systemcdes*, *systemaes* AND *wb_dma*

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec	diag	dtg
systemcdes	0	0	2	79	0	6	3.000	0.333	1.000	5	0
systemcdes	0	0	2	80	2	2	1.000	1.000	1.000	8	2
systemcdes	0	1	3	80	0	8	2.667	0.375	1.000	17	2
systemcdes	0	1	3	82	4	3	1.000	1.000	1.000	19	6
systemcdes	0	2	4	82	0	9	2.250	0.444	1.000	53	6
systemcdes	0	2	4	84	18	6	1.500	0.667	1.000	83	24
systemcdes	0	3	4	84	0	9	2.250	0.444	1.000	105	36
systemcdes	0	3	4	85	1	8	2.000	0.500	1.000	106	37
systemcdes	0	4	4	85	0	11	2.750	0.364	1.000	162	66
systemcdes	0	4	4	87	8	6	1.500	0.667	1.000	176	74
systemcdes	0	6	5	87	0	11	2.200	0.455	1.000	286	126
systemcdes	0	6	5	88	7	10	2.000	0.500	1.000	299	133
systemcdes	0	7	5	88	0	13	2.600	0.385	1.000	355	156
systemcdes	0	7	5	89	3	6	1.200	0.833	1.000	360	159
systemcdes	0	9	5	89	0	11	2.200	0.455	1.000	465	216
systemcdes	0	9	5	90	1	8	1.600	0.625	1.000	466	217
systemcdes	0	10	5	90	0	11	2.200	0.455	1.000	508	246
systemcdes	0	10	5	92	4	6	1.200	0.833	1.000	512	250
systemcdes	0	11	6	92	0	17	2.833	0.294	0.833	554	276
systemcdes	0	11	6	95	6	10	1.667	0.500	0.833	563	282
systemcdes	0	12	6	95	0	13	2.167	0.462	1.000	618	306
systemcdes	0	12	6	97	10	10	1.667	0.600	1.000	630	316
systemcdes	0	13	6	97	0	14	2.333	0.429	1.000	668	336
systemcdes	0	13	6	99	3	11	1.833	0.545	1.000	672	339
systemcdes	0	14	6	99	0	13	2.167	0.462	1.000	725	366
systemcdes	0	14	6	100	1	11	1.833	0.545	1.000	726	367
systemcdes	0	15	6	100	0	18	3.000	0.333	1.000	761	396
systemcdes	0	15	6	102	15	15	2.500	0.400	1.000	789	411
systemcdes	0	16	7	102	0	19	2.714	0.368	1.000	835	426
systemcdes	0	16	7	106	15	15	2.143	0.467	1.000	861	441
systemcdes	0	17	7	106	0	15	2.143	0.467	1.000	898	456
systemcdes	0	17	7	107	2	14	2.000	0.500	1.000	901	458
systemcdes	0	18	7	107	0	18	2.571	0.389	1.000	959	486
systemcdes	0	18	7	109	19	16	2.286	0.438	1.000	995	505
systemcdes	0	19	7	109	0	16	2.286	0.438	1.000	1025	516
systemcdes	0	19	7	111	27	14	2.000	0.500	1.000	1077	543
systemcaes	0	0	3	166	0	9	3.000	0.333	1.000	19	0
systemcaes	0	0	3	167	1	3	1.000	1.000	1.000	20	1
systemcaes	0	1	4	167	0	13	3.250	0.308	1.000	41	1
systemcaes	0	1	4	168	2	11	2.750	0.364	1.000	44	3
systemcaes	0	2	7	168	0	18	2.571	0.389	1.000	229	31
systemcaes	0	2	7	170	25	15	2.143	0.467	1.000	277	56
systemcaes	0	3	7	170	0	21	3.000	0.333	1.000	312	61
wb_dma	0	1	4	185	0	9	2.250	0.444	1.000	58	30
wb_dma	0	1	4	186	16	7	1.750	0.571	1.000	81	46
wb_dma	0	2	4	186	0	14	3.500	0.286	1.000	87	60
wb_dma	0	2	4	187	25	9	2.250	0.444	1.000	120	85
wb_dma	0	4	5	187	0	14	2.800	0.357	1.000	202	120
wb_dma	0	4	5	188	10	13	2.600	0.385	1.000	219	130
wb_dma	0	6	5	188	0	11	2.200	0.455	1.000	309	180
wb_dma	0	6	5	189	4	10	2.000	0.500	1.000	316	184
wb_dma	0	7	5	189	0	11	2.200	0.455	1.000	374	210
wb_dma	0	7	5	191	26	7	1.400	0.714	1.000	402	236
wb_dma	0	9	6	191	0	13	2.167	0.462	1.000	472	270
wb_dma	0	9	6	193	8	11	1.833	0.545	1.000	480	278
wb_dma	0	12	7	193	0	18	2.571	0.389	1.000	703	360

among the first defects it considers. Afterwards, even if the defects have an excess larger than two, the excess is smaller than for the first defects. This is significant since the later defects have higher multiplicities, and therefore, are likely to produce larger sets of candidate faults. The generation of new diagnostic tests counters this trend and ensures that smaller excess is obtained for defects with higher multiplicities.

The number of tests is increased to support the derivation of smaller sets of candidate faults. The increase starting from a diagnostic test set for fault pairs is typically smaller than 10%. Starting from a fault detection test set, the number of tests remains significantly lower than the number of tests in a diagnostic test set that targets fault pairs.

It is possible to continue considering additional defects, and generating additional diagnostic tests. However, the benefit from targeting additional defects decreases as more diagnostic tests are available.

B. New Defects

The results from the previous section indicate that it becomes more difficult to find defects with large sets of candidate faults as the procedure from Figure 1 generates more diagnostic tests. This indicates that the tests are useful for defects other than the ones targeted for diagnostic test generation. More direct evidence of this is given in this section.

TABLE IX
INITIAL FAULT DETECTION TEST SET, FIRST DEFECTS

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec	diag	dtg
s13207	0	1	2	468	0	11	5.500	0.182	1.000	40	30
s13207	0	1	2	469	11	10	5.000	0.200	1.000	49	41
s13207	0	3	2	469	0	15	7.500	0.133	1.000	102	90
s13207	0	3	2	471	21	6	3.000	0.333	1.000	104	111
s13207	0	4	2	471	0	12	6.000	0.167	1.000	108	111
s13207	0	4	2	472	30	11	5.500	0.182	1.000	129	141
s13207	0	9	3	472	0	14	4.667	0.214	1.000	320	261
s13207	0	9	3	473	24	13	4.333	0.231	1.000	337	285
s15850	0	6	3	379	0	14	4.667	0.214	1.000	172	163
s15850	0	6	3	380	1	13	4.333	0.231	1.000	173	164
b15	0	0	3	320	0	7	2.333	0.429	1.000	30	0
b15	0	0	3	322	12	5	1.667	0.600	1.000	34	12
b15	0	1	5	322	0	11	2.200	0.364	0.800	90	30
b15	0	1	5	323	2	10	2.000	0.400	0.800	93	32
b15	0	2	5	323	0	11	2.200	0.455	1.000	166	60
b15	0	2	5	324	4	10	2.000	0.500	1.000	171	64
b15	0	3	5	324	0	19	3.800	0.263	1.000	211	90
b15	0	3	5	325	6	16	3.200	0.312	1.000	222	96
b15	0	4	5	325	0	12	2.400	0.417	1.000	271	120
b15	0	4	5	326	3	10	2.000	0.500	1.000	274	123
b15	0	5	6	326	0	17	2.833	0.353	1.000	293	150
b15	0	5	6	328	8	10	1.667	0.600	1.000	307	158
b15	0	6	6	328	0	17	2.833	0.353	1.000	352	180
b15	0	6	6	331	30	14	2.333	0.429	1.000	407	210
b15	0	7	7	331	0	17	2.429	0.412	1.000	432	210
b15	0	7	7	332	3	15	2.143	0.400	0.857	435	213
b15	0	8	7	332	0	18	2.571	0.389	1.000	486	240
b15	0	8	7	335	9	14	2.000	0.500	1.000	497	249
b20	0	0	2	266	0	9	4.500	0.222	1.000	1	0
b20	0	0	2	268	20	4	2.000	0.500	1.000	15	20
b20	0	1	2	268	0	5	2.500	0.400	1.000	20	30
b20	0	1	2	269	5	3	1.500	0.667	1.000	29	35
b20	0	4	4	269	0	27	6.750	0.148	1.000	182	120
b20	0	4	4	270	14	5	1.250	0.600	0.750	209	134
b20	0	5	6	270	0	50	8.333	0.120	1.000	261	150
b20	0	5	6	271	9	43	7.167	0.140	1.000	274	159
b20	0	6	6	271	0	13	2.167	0.462	1.000	314	180
b20	0	6	6	272	10	12	2.000	0.500	1.000	331	190
b20	0	7	6	272	0	41	6.833	0.146	1.000	363	210
b20	0	7	6	274	11	6	1.000	0.833	0.833	377	221
b20	0	8	7	274	0	20	2.857	0.350	1.000	422	221
b20	0	8	7	275	18	14	2.000	0.500	1.000	443	239
b20	0	9	7	275	0	19	2.714	0.368	1.000	457	251
b20	0	9	7	276	16	8	1.143	0.750	0.857	482	267
aes_core	0	0	2	285	0	5	2.500	0.400	1.000	5	0
aes_core	0	0	2	286	3	4	2.000	0.500	1.000	6	3
aes_core	0	1	3	286	0	8	2.667	0.375	1.000	24	12
aes_core	0	1	3	287	2	7	2.333	0.429	1.000	27	14
aes_core	0	2	3	287	0	8	2.667	0.375	1.000	46	27
aes_core	0	2	3	289	13	3	1.000	1.000	1.000	64	40
tv80	0	2	5	706	0	11	2.200	0.455	1.000	152	54
tv80	0	2	5	707	3	9	1.800	0.556	1.000	157	57
tv80	0	3	5	707	0	13	2.600	0.385	1.000	194	84
tv80	0	3	5	708	16	12	2.400	0.417	1.000	209	100
tv80	0	4	5	708	0	12	2.400	0.417	1.000	230	114
tv80	0	4	5	710	13	10	2.000	0.500	1.000	238	127

For the experiment in this section, two test sets are considered for the circuits from Tables III and IV, the diagnostic test set T_{diag}, and the extended diagnostic test set obtained by applying the procedure from Figure 1 starting from T_{diag}. The resulting test set is denoted by T_{diag+}. T_{diag} is used as a starting point since it is the most suitable for logic diagnosis, and the most difficult to improve.

Up to 100 new defects are computed using the procedure from Section II, and logic diagnosis is applied using T_{diag}, followed by T_{diag+} to narrow down the set of candidate faults obtained with T_{diag}. The defects with the largest sets of candidate faults across all the circuits, for which the set of candidate faults is reduced, are reported in Table X.

TABLE X
INITIAL DIAGNOSTIC TEST SET, NEW DEFECTS

circuit	ts	ind	mult	tests	att	cand	excess	resol	prec
b11	1	24	7	187	0	31	4.429	0.161	0.714
b11	1+	24	7	203	0	23	3.286	0.217	0.714
b11	1	19	7	187	0	23	3.286	0.261	0.857
b11	1+	19	7	203	0	21	3.000	0.286	0.857
steppermotordrive	1	11	7	87	0	22	3.143	0.318	1.000
steppermotordrive	1+	11	7	99	0	18	2.571	0.333	0.857
b11	1	20	7	187	0	20	2.857	0.200	0.571
b11	1+	20	7	203	0	15	2.143	0.267	0.571
s1423	1	9	7	169	0	19	2.714	0.368	1.000
s1423	1+	9	7	196	0	18	2.571	0.389	1.000
b03	1	2	7	71	0	19	2.714	0.368	1.000
b03	1+	2	7	79	0	17	2.429	0.412	1.000
b04	1	0	7	139	0	19	2.714	0.368	1.000
b04	1+	0	7	140	0	18	2.571	0.389	1.000
b08	1	9	7	128	0	19	2.714	0.316	0.857
b08	1+	9	7	135	0	9	1.286	0.556	0.714
pci_spoci_ctrl	1	17	5	203	0	19	3.800	0.211	0.800
pci_spoci_ctrl	1+	17	5	205	0	18	3.600	0.222	0.800
s1423	1	3	5	169	0	18	3.600	0.278	1.000
s1423	1+	3	5	196	0	11	2.200	0.455	1.000
s1423	1	11	7	169	0	17	2.429	0.412	1.000
s1423	1+	11	7	196	0	14	2.000	0.500	1.000
b08	1	5	6	128	0	17	2.833	0.235	0.667
b08	1+	5	6	135	0	16	2.667	0.250	0.667
b11	1	10	6	187	0	17	2.833	0.294	0.833
b11	1+	10	6	203	0	13	2.167	0.385	0.833
b05	1	10	7	190	0	16	2.286	0.438	1.000
b05	1+	10	7	203	0	13	1.857	0.538	1.000
b08	1	12	7	128	0	16	2.286	0.438	1.000
b08	1+	12	7	135	0	15	2.143	0.467	1.000
b11	1	15	6	187	0	16	2.667	0.312	0.833
b11	1+	15	6	203	0	15	2.500	0.333	0.833
b11	1	18	7	187	0	16	2.286	0.438	1.000
b11	1+	18	7	203	0	15	2.143	0.467	1.000
s526	1	14	6	123	0	15	2.500	0.400	1.000
s526	1+	14	6	126	0	14	2.333	0.429	1.000
b05	1	11	7	190	0	15	2.143	0.467	1.000
b05	1+	11	7	203	0	13	1.857	0.538	1.000
b09	1	3	7	68	0	15	2.143	0.467	1.000
b09	1+	3	7	72	0	14	2.000	0.500	1.000
b11	1	23	7	187	0	15	2.143	0.400	0.857
b11	1+	23	7	203	0	14	2.000	0.429	0.857

From Table X it can be seen that T_{diag+} results in reduced sets of candidate faults for defects that were not considered during diagnostic test generation.

VI. CONCLUDING REMARKS

This paper observed that large sets of candidate faults are obtained when multiple defects are present in a faulty unit, even if a diagnostic test set is used for logic diagnosis. The paper analyzed the conditions that cause a large set of candidate faults to be formed under a logic diagnosis procedure that uses fault simulation to assign scores to modeled faults. Based on this analysis, the paper suggested new targets for diagnostic test generation. The targets are different from the indistinguished fault pairs typically targeted by diagnostic test generation procedures. Experimental results for benchmark circuits demonstrated that a diagnostic test set can be improved by adding tests for the new diagnostic targets.

REFERENCES

[1] D. B. Lavo, B. Chess, T. Larrabee and I. Hartanto, "Probabilistic Mixed-Model Fault Diagnosis", in Proc. Intl. Test Conf., 1998, pp. 1084-1093.
[2] S. Venkataraman and S. B. Drummonds, "POIROT: A Logic Fault Diagnosis Tool and its Applications", in Proc. Intl. Test Conf., 2000, pp. 253-262.

[3] T. Bartenstein, D. Heaberlin, L. Huisman and D. Sliwinski, "Diagnosing Combinational Logic Designs Using the Single Location At-A-Time (SLAT) Paradigm", in Proc. Intl. Test Conf., 2001, pp. 287-296.

[4] D. B. Lavo, I. Hartanto and T. Larrabee, "Multiplets, Models, and the Search for Meaning: Improving Per-Test Fault Diagnosis", in Proc. Intl. Test Conf., 2002, pp. 250-259.

[5] Y.-C. Lin and K.-T. Cheng, "Multiple-Fault Diagnosis Based on Single-Fault Activation and Single-Output Observation", in Proc. Design Automation & Test in Europe Conf., 2006, pp. 1-6.

[6] Z. Wang, M. Marek-Sadowska, K.-H. Tsai and J. Rajski, "Analysis and Methodology for Multiple-fault Diagnosis", IEEE Trans. on Computer-Aided Design, March 2006, Vol. 25, No. 3, pp. 558-575.

[7] S. Holst and H.-J. Wunderlich, "Adaptive Debug and Diagnosis without Fault Dictionaries", in Proc. European Test Symp., 2007, pp. 7-12.

[8] X. Yu and R. D. Blanton, "Multiple Defect Diagnosis Using No Assumptions on Failing Pattern Characteristics", in Proc. Design Autom. Conf., 2008, pp. 361-366.

[9] W.-T. Cheng, B. Benware, R. Guo, K.-H. Tsai, T. Kobayashi, K. Maruo, M. Nakao, Y. Fukui and H. Otake, "Enhancing Transition Fault Model for Delay Defect Diagnosis", in Proc. Asian Test Symp., 2008, pp. 179-184.

[10] V. J. Mehta, M. Marek-Sadowska, K.-H. Tsai and J. Rajski, "Timing-Aware Multiple-Delay-Fault Diagnosis", IEEE Trans. on Computer-Aided Design, Feb. 2009, pp. 245-258.

[11] M.-J. Tasi, M. C. T. Chao, J.-Y. Jou and M.-C. Wu, "Multiple-Fault Diagnosis Using Faulty-Region Identification", in Proc. VLSI Test Symp., 2009 pp. 123-128.

[12] X. Yu and R. D. Blanton, "Diagnosis of Integrated Circuits With Multiple Defects of Arbitrary Characteristics", IEEE Trans. on Computer-Aided Design, June 2010, Vol. 29, No. 6, pp. 977-987.

[13] X. Tang, W.-T. Cheng, R. Guo and S. M. Reddy, "Diagnosis of Multiple Physical Defects Using Logic Fault Models", in Proc. Asian Test Symp., 2010, pp. 94-99.

[14] J. Ye, Y. Hu, X. Li, W.-T. Cheng, Y. Huang and H. Tang, "Diagnose Failures Caused by Multiple Locations at a Time", IEEE Trans. on VLSI Systems, April 2014, Vol. 22, No. 4, pp. 824-837.

[15] I. Pomeranz, "OBO: An Output-By-Output Scoring Algorithm for Fault Diagnosis", in Proc. IEEE Computer Society Annual Symp. on VLSI, 2014.

[16] P.-J. Chen, C.-C. Che, J. C.-M. Li, S.-F. Kuo, P.-Y. Hsueh, C.-Y. Kuo and J.-N. Lee, "Physical-aware Systematic Multiple Defect Diagnosis", IET Computers & Digital Techniques, 2014, Vol. 8, No. 5, pp. 199-209.

[17] K. Gearhardt, C. Schuermyer and R. Guo, "Improving Fault Isolation Using Iterative Diagnosis", in Proc. Intl. Symp. for Testing and Failure Analysis, 2008, pp. 390-395.

[18] P. Camurati, D. Medina, P. Prinetto and M. Sonza Reorda, "A Diagnostic Test Pattern Generation Algorithm", in Proc. Intl. Test Conf., 1990, pp. 52-58.

[19] T. Gruning, U. Mahlstedt and H. Koopmeiners, "DIATEST: A Fast Diagnostic Test Pattern Generator for Combinational Circuits", in Proc. Intl. Conf. on Computer-Aided Design, Nov. 1991, pp. 194-197.

[20] F. Corno, P. Prinetto, M. Rebaudengo and M. Sonza Reorda, "GARDA: A Diagnostic ATPG for Large Synchronous Sequential Circuits", in Proc. Europ. Design & Test Conf., 1995, pp. 267-271.

[21] I. Pomeranz and S. M. Reddy, "A Diagnostic Test Generation Procedure for Synchronous Sequential Circuits Based on Test Elimination", in Proc. Intl. Test Conf., 1998, pp. 1074-1083.

[22] A. Veneris, R. Chang, M. S. Abadir and S. Seyedi, "Functional Fault Equivalence and Diagnostic Test Generation in Combinational Logic Circuits Using Conventional ATPG", Journal of Electronic Testing: Theory and Applications (JETTA), vol. 21, no. 5, Oct. 2005, pp. 495-502.

[23] R. Adapa, E. Flanigan and S. Tragoudas, "A Novel Test Generation Methodology for Adaptive Diagnosis", in Intl. Symp. on Quality Electronic Design, 2008, pp. 242-245.

[24] M. A. Shukoor and V. D. Agrawal, "A Two Phase Approach for Minimal Diagnostic Test Set Generation", in Proc. European Test Symp., 2009, pp. 115-120.

[25] Y. Higami, Y. Kurose, S. Ohno, H. Yamaoka, H. Takahashi, Y. Shimizu, T. Aikyo and Y. Takamatsu, "Diagnostic Test Generation for Transition Faults using a Stuck-at ATPG Tool", in Proc. Intl Test Conf., 2009, pp. 1-9.

[26] Y. Zhang and V. D. Agrawal, "A Diagnostic Test Generation System", in Proc. Intl. Test Conf, 2010, pp. 1-9.

[27] J. Ye, X. Zhang, Y. Hu and X. Li, "Substantial Fault Pair At-a-Time (SFPAT): An Automatic Diagnostic Pattern Generation Method", in Proc. Asian Test Symp., 2010, pp. 192-197.

[28] R. Guo. W.-T. Cheng, T. Kobayashi and K.-H. Tsai, "Diagnostic Test Generation for Small Delay Defect Diagnosis", in Proc. Intl Symp. on VLSI Design Autom. and Test, 2010, pp. 224-227.

[29] I. Pomeranz, "Generation of Mixed Broadside and Skewed-Load Diagnostic Test Sets for Transition Faults", in Proc. Pacific Rim Intl. Symp. on Dependable Computing, 2011, pp. 45-52.

[30] S. Prabhu, M. S. Hsiao, L. Lingappan and V. Gangaram, "A SMT-based Diagnostic Test Generation Method for Combinational Circuits", in Proc. VLSI Test Symp., 2012 pp. 215-220.

[31] Y.-H. Chen, C.-L. Chang and C.H.-P. Wen, "Diagnostic Test-pattern Generation Targeting Open-segment Defects and its Diagnosis Flow", IET Computers & Digital Techniques, Vol. 6, No. 3, 2012, pp. 186-193.

[32] M. Scholzel, T. Koal and H. T. Vierhaus, "Systematic Generation of Diagnostic Software-based Self-test Routines for Processor Components", in Proc. European Test Symp., 2014, pp. 1-6.

[33] Z. Sun, A. Bosio, L. Dilillo, P. Girard, A. Virazel and E. Auvray, "On the Generation of Diagnostic Test Set for Intra-cell Defects", in Proc. Asian Test Symp., 2014, pp. 312-317.

[34] A. Riefert, R. Cantoro, M. Sauer, M. Sonza Reorda and B. Becker, "Effective Generation and Evaluation of Diagnostic SBST Programs", in Proc. VLSI Test Symp., 2016, pp. 1-6.

[35] M. E. Amyeen, D. Kim, M. Chandrasekar, M. Noman, S. Venkataraman, A. Jain, N. Goel and R. Sharma, "A novel Diagnostic test Generation Methodology and its Application in Production Failure Isolation", in Proc. Intl. Test Conf., 2016, pp. 1-10.

[36] M. E. Amyeen, I. Pomeranz and S. Venkataraman, "A Joint Diagnostic Test Generation Procedure with Dynamic Test Compaction", in Proc. Asian Test Symp., 2016.

[37] I. Pomeranz, "Improving the Accuracy of Defect Diagnosis by Considering Fewer Tests", IEEE Trans. on Computer-Aided Design, Dec. 2014, pp. 2010-2014.

[38] I. Pomeranz, "A Test Selection Procedure for Improving the Accuracy of Defect Diagnosis", IEEE Trans. on VLSI Systems, Aug. 2016, pp. 2759-2767.

[39] N. Wang, I. Pomeranz, B. Benware, M. E. Amyeen and S. Venkataraman, "Improving the Resolution of Multiple Defect Diagnosis by Removing and Selecting Tests", in Proc. Intl. Symp. on Defect and Fault Tolerance in VLSI and Nanotechnology Systems, Oct. 2018.

PLACE PHOTO HERE

Irith Pomeranz (M'89-SM'96-F'99) received the B.Sc degree (Summa cum Laude) in Computer Engineering and the D.Sc degree from the Department of Electrical Engineering at the Technion - Israel Institute of Technology in 1985 and 1989, respectively. From 1989 to 1990 she was a Lecturer in the Department of Computer Science at the Technion. From 1990 to 2000 she was a faculty member in the Department of Electrical and Computer Engineering at the University of Iowa. In 2000 she joined the School of Electrical and Computer Engineering at Purdue University. Her research interests include testing of VLSI circuits, design for testability, synthesis and design verification. Dr. Pomeranz is a recipient of the NSF Young Investigator Award in 1993, and of the University of Iowa Faculty Scholar Award in 1997. Three of her conference papers won best paper awards, and four other papers were nominated for best paper awards. One of the papers she co-authored was selected by the 2016 International Test Conference as the most significant paper published ten years before. She delivered a keynote speech at the 2006 Asian Test Symposium. She served as associate editor of the ACM Transactions on Design Automation, the IEEE Transactions on Computers, and the IEEE Transactions on VLSI Systems. She served as guest editor of the IEEE Transactions on Computers January 1998 special issue on Dependability of Computing Systems, and as program co-chair of the 1999 Fault-Tolerant Computing Symposium. She served as program chair of the 2004 and 2005 VLSI Test Symposium, and as general chair of the 2006 VLSI Test Symposium. She is a Golden Core Member of the IEEE Computer Society.