Diagnosing Realistic Bridging Faults with Single Stuck-at Information

David B. Lavo*, *Student Member, IEEE* Brian Chess†, *Member, IEEE*
Tracy Larrabee*, *Senior Member, IEEE* F. Joel Ferguson*, *Member, IEEE*

Abstract

Successful failure analysis requires accurate fault diagnosis. This paper presents a method for diagnosing bridging faults that improves on previous methods. The new method uses single stuck-at fault signatures, produces accurate and precise diagnoses, and takes into account imperfect fault modeling; it accomplishes this by introducing the concepts of *match restriction*, *match requirement*, and *match ranking*.

1 Introduction

Rigorous testing of circuits can prevent the shipment of defective parts, but improving the production quality of a circuit depends upon effective *failure analysis*, the process of determining the cause of detected failures. Discovering the cause of failures in a circuit can often lead to improvements in circuit design or the manufacturing process, with a subsequent production of higher-quality integrated circuits.

Failure analysis usually consists of two tasks. The first, *fault diagnosis*, is a logical search to determine the likely sources of error, using circuit information and details about how the circuit failed. The second, *fault location* or *defect identification*, is a physical search to discover the mechanism of failure in the actual defective part. Given the enormous number of circuit elements in modern ICs, and the number of layers in most complex circuits, physical search cannot succeed without considerable guidance from fault diagnosis. If the diagnosis is either inaccurate or imprecise (identifying either incorrect or excessively many fault candidates, respectively), the process of fault location will consume, and possibly waste, considerable amounts of time and effort.

This paper presents a technique for the accurate and precise diagnosis of *bridging faults*, originally defined by Mei as the unintentional electrical shorting of two gate outputs [32]. The scope of

*Current address is Department of Computer Engineering, University of California, Santa Cruz 95064. This work was supported by the Semiconductor Research Corporation under Contract 93-DJ-315 and the National Science Foundation under grant MIP-9011254.

†When this work was performed, Brian Chess was with the Department of Computer Engineering, UCSC. His current address is Hewlett-Packard Company, 1501 Page Mill Road MS-6UJ, Palo Alto CA 94304

the problem is limited to combinational CMOS circuits; *full-scan* sequential circuits, in which all state elements are controllable and observable, may be considered combinational for purposes of test and diagnosis. In addition, tests on a circuit are limited to examining logic values at the circuit outputs. The proposed technique is demonstrated with simulated bridging faults on the ISCAS-85 benchmark circuits [10]. The results given indicate the success of the technique; diagnoses are of very high quality, both accurate and precise, and are often ideal: the correct identification of a single pair of shorted circuit nodes.

Section 2 of this paper presents background material on fault diagnosis, and bridging fault diagnosis in particular. Section 3 presents one technique for diagnosing bridging faults using single stuck-at fault signatures. Section 4 gives an experimental analysis of the original technique, presenting baseline results. Section 5 introduces and describes an improved diagnostic technique, and demonstrates the improved results obtained. Section 6 compares the improved technique to another diagnosis method that employs a different approach to the problem and demonstrates the advantages of the proposed method, which leads into the conclusions in Section 7.

2 Background

This section contains a brief discussion of the fundamentals of diagnosis and introduces the terms involved. It describes the general approaches to the basic problem of fault diagnosis, and then presents several previously-proposed diagnosis techniques, distinguished by the approach taken and the type of fault targeted.

2.1 Diagnosis and fault models

As stated in the introduction, fault diagnosis is a component of failure analysis; appropriately, its domain is that of the *logic fault*, or simply *fault*, which is an abstract representation of how an element in a defective circuit misbehaves. A description of the behavior and assumptions about the nature of a logic fault is referred to as a *fault model* [1].

As with testing, diagnosis traditionally involves the choice of a fault model; the most popular fault model for both testing and diagnosis is the *single stuck-at fault model*, in which a node in the circuit is assumed to be unable to change its logic value. The stuck-at model is popular due to its simplicity, and because it has proven to be effective both in providing high defect coverage when used as a fault model for test generation and when diagnosing a limited range of faulty behaviors [26]. Other fault models can be used in diagnosis, as will be discussed shortly.

The concepts of fault and fault model are separate from that of a *defect*, which usually refers to the physical mechanism, such as an electrical short or open, that produces the incorrect behavior of the circuit. Feltham and Maly demonstrated that many defects in modern CMOS technologies cause changes in the circuit description that result in electrical shorts [19], which implies that many failures can be modeled by bridging faults [32]. Intuitively, identifying a fault as the cause of a defect has much to do with the relative likelihood of certain defects occurring in the actual circuit. *Inductive Fault Analysis* [40] uses the circuit layout to determine the relative probabilities of individual physical faults in the fabricated circuit.

Inductive fault analysis (IFA) uses the concept of a *spot defect* (or *point defect*), which is an area of extra or missing conducting material that creates an unintentional electrical short or break in a circuit. As these spot defects often result in bridge or open faults, inductive fault analysis can provide a fault diagnosis of sorts: an ordered list of physical faults (bridges or opens) that are likely to occur, in which the order is defined by the relative probability of each associated fault. The relative probability of a fault is expressed as its *weighted critical area* (WCA), defined as the physical area of the layout that is sensitive to the introduction of a spot defect, multiplied by the defect density for that defect type. For example, two circuit nodes that run close to one another for a relatively long distance provide a large area for the introduction of a shorting point defect; the resulting large WCA value indicates that a bridging fault between these nodes is considered relatively likely.

Inductive fault analysis can alternatively be applied to diagnosis for the creation of fault lists. Inductive fault analysis tools such as Carafe [24, 25] can provide a *realistic fault list*, important for fault models such as the bridging fault model, in which the number of possible faults is intractable for most circuits. By limiting the candidates to only faults that can realistically occur in the fabricated circuit, a diagnosis can be obtained that is much more precise than one that results from consideration of all theoretical faults. Section 4 explains how Carafe was used in the proposed technique for exactly this purpose.

While it is common (and convenient) to speak of diagnosis identifying or locating faults in a circuit, the underlying target of diagnosis is ultimately a physical defect; the fault models used are simply useful abstractions in the eventual identification of a defect or defect location. As will be discussed in the following sections, the association of diagnostic fault model to targeted defect is not inviolable: a diagnosis may be performed using one fault model while targeting a defect more accurately represented by another fault model.

The traditional method of fault diagnosis, referred to as *cause-effect analysis* by Abramovici *et al.* [1], has been described by Aitken [7] as *test-based fault localization*: identification of a defect location by comparing failures observed on a tester with those predicted by fault simulation. A fault simulator describes the behavior of a circuit in the presence of a fault, usually in the form of a *fault signature*. A fault signature is the complete list of all input patterns (or *test vectors*) and circuit outputs by which a fault is detected.[1] The process of test-based fault localization is one of comparing the observed faulty behavior of the circuit with a set of fault signatures, each representing a fault candidate. The resulting set of matches, if any, constitutes a diagnosis.

Many early diagnostic systems used a simple matching process, in which the signature of a fault candidate would either have to match exactly the circuit's fault signature, containing every

[1] Some authors have reserved the term **fault signature** for only the response of faulty circuits under test. Breuer [9] defines a fault signature as "the characteristic function of the erroneous response produced by ... [a] fault", without regard to fault type. In this paper, as in much of common usage, the term **signature** is applied to actual behaviors, as well as simulated and abstract faults, as in **stuck-at signature** and **composite signature**, introduced later.

error-carrying vector and output, or would have to be a subset thereof. As diagnostic techniques matured, the matching process became more flexible; a good example of a simple generalization is the partial-intersection operation presented by Kunda [28] that ranked matches by the size of intersection. Subsequent sections of this paper will show that the matching algorithms employed by diagnostic techniques are often essential in translating from abstract fault models to defects, or from targeted fault models to untargeted faults, or to handle the vagaries of faulty circuit behavior.

Most traditional (cause-effect) techniques involve two primary elements: a fault model, and a comparison or matching algorithm. [2]

2.2 Stuck-at fault diagnosis

Early fault diagnosis systems targeted only stuck-at faults; the fault candidates were stuck-at nodes, and the candidates were described by stuck-at fault signatures. In addition, the actual defect mechanism was interpreted strictly as a single stuck-at circuit node; other defect types could necessarily not be precisely diagnosed.

Many early systems of VLSI diagnosis, such as Western Electric Company's DORA [34] and an early approach of Teradyne, Inc. [36], attempted to incorporate the concept of test-based fault localization with the previous-generation method of diagnosis, called *guided-probe analysis*. Guided-probe analysis employed a physical voltage probe and feedback from an analysis algorithm to intelligently select accessible circuit nodes for evaluation. The Teradyne and DORA techniques attempted to supplement the guided-probe analysis algorithm with information from stuck-at signatures.

Both systems used relatively advanced (for their time) matching algorithms. The DORA system used a nearness calculation that the authors describe as *fuzzy match*. The Teradyne system employed the concept of *prediction penalties*: the signature of a candidate fault is considered a prediction of some faulty behavior, made up of $\langle \text{output} : \text{vector} \rangle$ pairs. When matching with the actual observed behavior, the Teradyne algorithm scored a candidate fault by penalizing for each $\langle \text{output} : \text{vector} \rangle$ pair found in the stuck-at signature but not found in the observed behavior, and penalizing for each $\langle \text{output} : \text{vector} \rangle$ pair found in the observed behavior but not the stuck-at signature. These have commonly become known as *misprediction* and *non-prediction* penalties, respectively. A related Teradyne system [37] introduced the processing of *possible-detects*, or outputs in stuck-at signatures that have unknown logic values, into the matching process.

A system that uses a more sophisticated algorithm of parameterized matching has been presented by De and Gunda [18]; in this system, the user can specify the relative importance of misprediction and non-prediction. A quantitative ranking is assigned to each stuck-at fault, from which some indication can be made about the existence of multiple stuck-at faults or possibly some unmodeled (non-stuck-at) faults. An even more complicated and rigorous approach to fault diagnosis has been suggested by Sheppard and Simpson [42, 41]; adapting concepts from machine learning, they propose applying methods of evidential reasoning to the failure data and dictionary

[2] An analysis of the interaction between the two elements of traditional diagnosis—fault model and matching algorithm—has been presented by Aitken and Maxwell [8].

information in an attempt to overcome noisy data and multiple fault conditions.

The example systems described above characterize the general trend of stuck-at model diagnosis, from simple to complex matching algorithms. It has become evident that many failures in CMOS circuits do not behave exactly like single stuck-at faults [7]. The inclusion of increasingly more-complicated algorithms is the necessary result of the reliance of these systems on the overly-simple single stuck-at fault model.

2.3 Bridging fault diagnosis

Several approaches have been taken towards incorporating the bridging fault model into traditional test-based fault localization. The first steps retained the legacy of stuck-at signatures, using these readily-available fault descriptions to approximate or identify bridging fault behavior. Many simple approaches, such as the Teradyne-based [37] system demonstrated in Section 6, merely compared stuck-at signatures to the observed behavior, and implicated the (single) nodes that most closely matched. A novel approach was presented by Millman, McCluskey, and Acken [33], in which pseudo-signatures for bridging faults were constructed from stuck-at signatures for the bridged nodes. The matching algorithm was simple subset matching; this technique is described in detail in Section 3. A more conventional application of stuck-at signatures, paired with a sophisticated matching algorithm, was presented by Chakravarty and Gong [13]. These last two methods suffer from imprecision: the average diagnoses for both methods are very large, consisting of hundreds or thousands of candidates.

Diagnosing bridging faults with available single stuck-at fault information is an appealing idea, but this approach can lead to unusably large diagnoses or an unacceptable percentage of *misleading diagnoses*, in which neither node involved in the actual short is identified by the fault candidates. To address these deficiencies, Aitken and Maxwell built dictionaries comprised of realistic faults [8]. This approach is truly cause-effect analysis using the bridging fault-model: the fault candidates are the same faults targeted for diagnosis. The results presented by the authors indicate both excellent accuracy and precision: there are very few misleading diagnoses, and the resulting diagnoses are very small (less than 10 candidates).

While there are obvious advantages to this approach, there are significant costs. The number of realistic faults in a circuit is significantly larger than the number of single stuck-at faults for a circuit; also, the cost of simulating each individual realistic fault is frequently much greater, requiring much more detailed knowledge of the circuit for model construction. In addition, actual bridging fault behavior often diverges from simulated behavior, requiring validation and refinement of the models [7]. The continued search for a method of diagnosing bridging faults using inexpensive stuck-at signatures is driven by the cost and complexity of realistic fault models; this paper presents such an approach, yielding similar results to the realistic fault model approach, but at a much lower cost.

A completely different approach is taken by methods referred to as I_{DDQ} *diagnosis* [5, 6, 14]. In $\mathrm{I_{DDQ}}$ diagnosis, an otherwise static circuit is monitored for excessive current flow, which would indicate a fault-induced path from power to ground. Fault signatures can be constructed for I_{DDQ} measurement; errors are detected at a single output, the point of current measurement. Normal test-based fault localization can then proceed, matching expected failures to observed failures. In addition, voltage (logic) measurements can be taken at the outputs, and conventional fault signatures used to refine the diagnosis. The advantages of I_{DDQ} diagnosis are that pass/fail I_{DDQ} signatures are easy to construct, and when I_{DDQ} diagnosis works, it seems to work well: the resulting diagnoses are usually both precise and accurate. The disadvantages are that not all circuits are I_{DDQ} testable; in addition, a large number of chips fail all I_{DDQ} patterns applied, a behavior that cannot be diagnosed. There is also a question of the effectiveness of I_{DDQ} testing and diagnosis given the current trend of generally increasing, and increasingly noisy, background current levels. A proposed solution to this problem is the concept of current signatures [11, 21, 22, 43], which considers differential current levels between tests, rather than absolute values, to infer about the presence and identity of circuit defects.

2.4 Other approaches

Several approaches to fault diagnosis are not neatly categorized by the combination of fault model and algorithm specification used above. Some have attempted to eliminate or minimize fault simulation, instead relying on such information as the propagation and sensitization cones of individual faults or fault-free circuit nodes. The approaches suggested by Abramovici and Breuer [9] and Rajski and Cox [35] are examples, and are referred to as *effect-cause analysis*. Both attempt to identify all fault-free lines, and so can implicitly diagnose multiple faults and various fault types, although the resulting diagnoses are often pessimistic and imprecise.

Waicukauski and Lindbloom present a technique that incorporates elements of both test-based fault localization and effect-cause analysis [44]. The technique relies on a great deal of information: in addition to propagation and sensitization path information, it requires knowledge of internal-node logic values to eliminate candidate nodes. Stuck-at fault simulation is performed, but only for a reduced set of fault candidates. While the presented theory assumes stuck-at behavior for individual faulty nodes on a per-vector basis, it also allows for complex fault behaviors, specifically multiple-site faults. While this technique offers a great deal of flexibility in targeting faults, its computational cost and diagnostic precision are matters of concern.

3 The MMA algorithm

This paper investigates and improves a bridging fault diagnosis technique using the single stuck-at fault model that was proposed by Millman, McCluskey, and Acken (henceforth called the MMA technique) [33]. The MMA technique has many advantages, the most notable of which are the use of the ubiquitous single stuck-at fault model, obviation of the need for additional circuit information for bridging fault diagnosis, and a small likelihood of misleading diagnoses under modeled conditions.

On the other hand, the MMA technique, like many other stuck-at based techniques, has the disadvantage of unusably large diagnoses. The MMA technique also disregarded bridge resistance, variable downstream logic thresholds, and the possibility of state-holding bridging fault behavior. These simplifying assumptions, however, enabled an approach to diagnosing bridging faults using relatively simple stuck-at information, a desirable feature considering the expense of realistic fault models.

3.1 MMA theory

The MMA diagnostic theory, described below, followed from some relatively simple observations about bridging fault behavior under the *voting model* [2]. If Vector v detects a bridging fault in a CMOS circuit, the two bridged nodes necessarily have opposite fault-free logic values when v is applied. The driving transistor networks of these two nodes will each attempt to assert competing logic values on the bridge. The application of v causes the stronger node to outvote the other, driving the outvoted node to a faulty logic value. The key observation of the MMA technique is that since v is able to sensitize the outvoted node and propagate the faulty value to a circuit output, it must also detect the stuck-at fault for the outvoted node stuck at the faulty value. Therefore, v must appear in a complete list of detecting vectors for the stuck-at fault on the outvoted node.

This complete list of the detecting vectors for a particular fault is contained in its fault signature. The basis of the MMA technique is the construction and use of *composite signatures* for each potential bridging fault: the composite signature of a bridging fault is the union of the four associated single stuck-at signatures (Figure 1). By the reasoning given above, MMA concludes that the fault signature of a bridging fault will be contained in, or will be a subset of, the bridging fault's composite signature.

XSA0	XSA1	YSA0	YSA1

Figure 1: The MMA composite signature for node X bridged to node Y is the union of the four stuck-at signatures for the two bridged nodes. Each stuck-at signature is a set of output:vector pairs.

The process of diagnosis can be outlined with the use of a few definitions. First, let v and o represent vector and output (single output pin) variables, respectively. Then, let ν represent a logic value: $\nu(o,v)$ is the logic value at output o upon application of vector v in the fault-free circuit; and $\nu_f(o,v)$ is the logic value at output o upon application of vector v in the presence of fault f.

The observed faulty behavior is represented by $\mathbf{B_f}$, the set of error-carrying $\langle$output : vector$\rangle$ pairs:

$$\mathbf{B_f} = \{\forall\langle o : v\rangle \mid \nu(o,v) \neq \nu_f(o,v)\}. \tag{1}$$

Fault	Vector(s)
Asa0	2, 4
Asa1	1
Bsa0	2
Bsa1	4, 5
Csa0	4
Csa1	1, 3

Table 1: Stuck-at signatures

Fault	Vector(s)
A @ B	1, 2, 4, 5
B @ C	1, 2, 3, 4, 5
C @ A	1, 2, 3, 4

Table 2: Composite signatures

In the rest of this description, the subscript $\mathbf{f}$ will be dropped from $\mathbf{B}$, since it is understood that $\mathbf{B}$ will refer to a single faulty behavior.

The MMA technique builds a composite signature, denoted here by $\mathbf{C_f}$, for every possible node pair in the circuit, from four stuck-at signatures, denoted by $\mathbf{S_f}$. In this notation, $\mathbf{S_{X0}}$ refers to the stuck-at signature for node X stuck-at 0, and $\mathbf{C_{X@Y}}$ refers to the composite signature for node X bridged to node Y. Duplicate entries in $\mathbf{C_f}$ are dropped after concatenation.

$$\mathbf{S_{X0}} = \{\forall\langle o : v\rangle \mid \nu(o,v) \neq \nu_{X0}(o,v)\} \tag{2}$$

$$\mathbf{C_{X@Y}} = \mathbf{S_{X0}} \bigcup \mathbf{S_{X1}} \bigcup \mathbf{S_{Y0}} \bigcup \mathbf{S_{Y1}} \tag{3}$$

The MMA diagnostic algorithm compares each composite signature with the observed behavior; a composite signature containing entries that are a superset of the entries contained in the observed faulty behavior is said to be a *match*.[3] The MMA diagnosis of a bridging fault is a list of candidate bridging faults having composite signatures that match the observed faulty behavior. A diagnosis can be formalized as

$$\mathbf{D} = \{\forall \mathbf{C_i} \mid \mathbf{B} \subseteq \mathbf{C_i}\}, \tag{4}$$

where the subscript $\mathbf{i}$ indicates an index through all (composite signature) candidates. To reiterate, this technique of bridging fault diagnosis does not require explicit simulation of bridging faults, only stuck-at fault simulation to create stuck-at signatures.

All of the previously-described operations, including composite signature construction and candidate matching, are demonstrated in Tables 1 and 2. Table 1 gives a stuck-at fault dictionary, or list of all stuck-at faults and their signatures, for a trivial circuit of only three nodes and a single output. The nodes are labeled A, B, and C; the output is unnamed and for simplicity is omitted from the signatures. Table 2 shows the resulting MMA composite bridging fault dictionary, with composite signatures constructed as described previously.

As an example, the observed behavior {1, 2, 4} is a subset of all three composite signature candidates; the diagnosis therefore implicates every node pair in the circuit and matches all three

[3]Note that since o represents a single output pin, there is an entry representing every error-carrying output pin for every vector in $\mathbf{S_f}$, $\mathbf{C_f}$, and $\mathbf{B_f}$. Therefore, the subset matching criteria applies to outputs as well as vectors; Section 4 contains a discussion of allowing matches on subsets of outputs versus requiring a match on the full output response for a detecting vector.

bridging faults. Observed behavior {2, 3} also poorly distinguishes between the available nodes and matches with faults B@C and C@A. The two diagnosis examples, while trivial, demonstrate both the relative simplicity and the imprecision characteristic of the MMA technique. Note that while the bridging fault diagnoses are imprecise, all six stuck-at faults are uniquely identified by their stuck-at signatures; this resolution is lost, however, in the construction of the composite signatures.

3.2 Evaluating diagnoses

If the identity of the actual fault is known, an individual match can be evaluated for correctness: does a matching composite signature identify, completely or partially, the nodes involved in the actual bridging fault? Extending this, the quality of a diagnosis can be evaluated as the quality of its component composite signature, or fault candidate, matches.

For this research, a bridging fault is assumed to involve exactly two nodes; all matching composite signatures similarly correspond to bridging fault candidates made up of two nodes. There are then three types of matches, as defined in the MMA paper: A *correct* match correctly identifies both of the nodes involved in the bridge, a *partial* match correctly identifies only one of the nodes involved in the bridge, and a *misleading* match identifies neither of the nodes involved in the bridge. An example bridging fault and examples of each type of match are given in Table 3.

Fault	Matching Candidate	Match Type
X @ Y	X @ Y	Correct
X @ Y	X @ Z	Partial
X @ Y	W @ Z	Misleading

Table 3: Example matches (fault to candidate), and corresponding match types.

Having defined the types of matches, in MMA terminology the quality of a diagnosis is indicated in one of three ways, based on the matches used to construct it. An *exact* diagnosis contains only the correct match, a *partial* diagnosis contains the correct match in addition to other matches, and an *incorrect* diagnosis does not contain the correct match. Incorrect diagnoses can be further divided into three categories: An *incomplete* diagnosis contains partial matches but not the correct match, a *misleading* diagnosis contains only misleading matches, and a *failed* diagnosis is empty. Although all types of incorrect diagnoses are undesirable, it is much better to have a failed diagnosis than a misleading diagnosis; a failed diagnosis is clearly incorrect and cannot mislead the search for a defect.

If a bridging fault can create a feedback loop in the circuit, some test vectors may cause circuit outputs to oscillate. Such a vector is said to only *possibly detect* (or *potentially detect*) the bridging fault. Under the assumptions made by the MMA paper,[4] the inclusion of possibly detecting test

[4]A restatement of the MMA assumptions: Two bridged nodes, with one node outvoted and error-carrying for any detecting vector; zero-resistance bridges; the bridge voltage is a definite logic value with regard to downstream gates.

Fault	Diagnosis (Matching Candidates)	Diagnosis Type
X @ Y	{X @ Y}	Exact
X @ Y	{X @ Y, X @ Z, W @ Z}	Partial
X @ Y	{X @ Z, W @ Z}	Incomplete
X @ Y	{W @ Z}	Misleading
X @ Y	{ }	Failed

Table 4: Diagnosis types. Incomplete, misleading, and failed diagnoses are all considered incorrect diagnoses, as they do not contain the correct match.

vectors can lead to misleading and failed diagnoses; but if the possibly detecting vectors are ignored, misleading and failed diagnoses will not occur, and the correct fault will always be part of the diagnosis. This is stated as a theorem in the MMA paper:

> When possibly detecting patterns are ignored, the fault signature of a bridging fault must be contained in its composite signature.

The MMA theorem guarantees that incorrect diagnoses will not occur, but it places no bound on the size of the diagnosis. Physical investigation of the failed part sometimes requires destruction of the layers above the site of the suspected defect, and once these layers are gone, nearby suspected sites cannot be investigated. Section 4.2 shows that the size of the average diagnosis using the MMA technique on the ISCAS-85 benchmark circuits [10], for these experiments, is at least 33 matches (for the C880) and can reach over 200 matches (for the C7552).

The MMA theorem guarantees that the correct match will appear in the diagnosis—as long as the observed behavior of the fault is not affected by variable logic thresholds, which commonly affect the behavior of faulty CMOS circuits [17, 31]. One of the goals of this research was to investigate the effects of such thresholds on the MMA technique, and ultimately to attempt to compensate for such effects in an improved diagnostic technique.

3.3 The Byzantine Generals Problem for Bridging Faults

In order to be detected with a logic test, a bridging fault must create an error that is propagated to one or more circuit outputs. At the fault site, this error is a voltage that is subject to interpretation as different logic values by downstream logic gates. Because gate input logic thresholds are not identical, different downstream gates can interpret the voltage as different logic values: this phenomenon is known as the Byzantine Generals Problem [29] for bridging faults [3, 4]. Figure 2 shows a simple example of voltage interpretation in the presence of variable logic thresholds.

This behavior has important implications for diagnosis: the propagation conditions for the error induced by a bridging fault are not necessarily the same as those caused by a stuck-at fault. Therefore, a detecting test vector may or may not display the same behavior at the circuit outputs for a bridging fault as for a stuck-at fault on one of the bridged nodes. Note that the faulty voltage

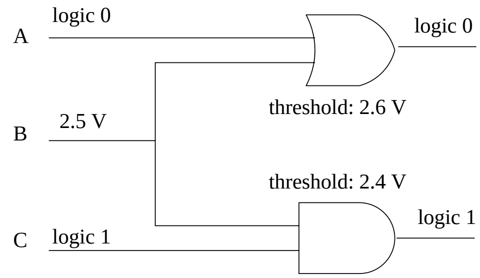

Figure 2: An instance of the Byzantine Generals Problem for bridging faults. Node B has an intermediate (faulty) voltage value due to the presence of a bridge. Each gate interprets the intermediate voltage as a different logic value.

on node B in Figure 2 will not cause the circuit to appear, for the node values shown, as if B were stuck-at 1 or stuck-at 0—each circuit output reports evidence that there is a different value on node B.

The Byzantine Generals problem can affect diagnosis in several ways. It might cause the error introduced by the bridging fault to be propagated to more or fewer circuit outputs than would be affected by a single stuck-at fault. Alternatively, the error introduced by the bridging fault may be detected by a vector, or propagated to an output, that would never evince an error for any one of the four single stuck-at faults. The Byzantine Generals Problem may also cause errors to occur downstream from both of the bridged nodes at the same time. Each of these is an example of how variable logic thresholds nullify the previously-stated MMA theorem and cause incorrect diagnoses to result for real circuits. Section 4.2 shows that up to 10% of the diagnostic trials resulted in incorrect diagnoses because of the Byzantine Generals Problem.

4 Experimental considerations

Before reporting on the quality of the original MMA technique as applied to the ISCAS-85 circuits [10], this section describes the creation of composite signatures, and the selection of faulty circuits to be considered in diagnosis experiments. It also describes the acquisition of faulty circuit behaviors, the creation of the initial stuck-at signatures, and an important distinction in the interpretation of the MMA theorem.

Since stuck-at signatures are used to create the MMA composite signatures, the diagnostic ability of the technique is limited by the diagnostic ability or quality of the stuck-at signatures. Previous attempts to measure the diagnostic quality of a given test set have usually involved calculations of the number of indistinguishable (bit-wise equivalent) fault signatures [12, 27, 38, 39]. A test set for which many faults have identical signatures has poor diagnostic resolution, as the likelihood of a precise diagnosis is small. Conversely, if a test set produces a unique signature for each fault, it is considered to have ideal diagnostic resolution. Diagnostic test pattern generators (DTPG) are designed to address this issue by creating test sets with as high a diagnostic resolution as possible.

Note that the concept of diagnostic resolution is only completely applicable to modeled faults; the connection to unmodeled faults is often tenuous or misleading. For purposes of this research, however, a strong correlation does exist. If composite signatures are used for diagnosis, indistinguishable stuck-at faults will result in indistinguishable bridging faults: if two nodes have identical stuck-at signatures, all corresponding composite signature involving these nodes will also be identical.

For this research, a diagnostic test set was generated by a DTPG system to assure the best possible stuck-at signatures and diagnosis information [15, 23]. The size of the test sets ranged from 60 vectors for the C6288 to 365 vectors for the C7552.

Given the stuck-at signatures, how are the composite signatures constructed? As described, a single entry is created from stuck-at signatures by concatenating the four fault signatures of two distinct nodes. There are, however, $\binom{n}{2}$ possible bridging faults in an n-node circuit, making both the construction and use of all possible composite signatures impractical for most circuits.

One solution is to limit the construction of composite signatures to only realistic bridging faults, as identified by inductive fault analysis (IFA).[5] IFA and realistic faults are described in Section 1 of this paper. For this research, realistic bridging fault lists were created for the MCNC layouts of the ISCAS-85 circuits using the program Carafe [24, 25]. Restricting the composite signatures to realistic bridging faults cuts the fault lists to a manageable length (for the ISCAS-85 circuits there are three to eight times as many realistic bridging faults as single stuck-at faults [20]).

For the diagnostic trials, faulty output responses were created for the 10% of realistic bridging faults that are most likely to occur based on layout and defect density information[6]. The faults were sampled to reduce the number of trials to a feasible number; the high-probability faults were considered statistically most interesting since, for a sample size of n, they are those n faults expected to occur the most often over a large number of diagnoses. Note that although only a sample of the realistic faults were simulated and diagnosed, the process of diagnosis considers all realistic faults as potential candidates.

Having determined which faults are to provide the faulty behaviors, the Nemesis bridging fault simulator [16, 30] provided the faulty signatures used to evaluate the MMA technique. Bridging fault simulation is used only to determine faulty output responses to be diagnosed and not as part of the diagnostic procedure. The behavioral model used by Nemesis for bridging faults is *two-component simulation*, in which the gates driving the bridged nodes are SPICE-simulated to determine the bridge voltage. The bridge voltage is then compared against the SPICE-computed

[5]While IFA requires layout information, this additional requirement is usually not unreasonable. Most often, before a defect is diagnosed, the physical design is completed and known, and therefore the realistic faults can be extracted.

[6]While this is obviously an interesting set of bridging faults, it is also a biased sample. To remedy this, two additional samples were taken of the realistic fault list. First, the *least-likely* 10% of the faults were simulated and diagnosed. Second, a *random* sample of 10% was drawn from the complete fault list, simulated and diagnosed. There is no significant difference in the results for the three fault samples.

logic thresholds of downstream gates in order to model the effects of the Byzantine Generals Problem.

The simulator also extensively models feedback bridging faults. If a feedback bridging fault evinces the potential to oscillate or hold state, the simulator biases the bridge voltage in favor of the fault-free value on the rear bridged node, thereby disallowing oscillation and state-holding behavior. This approximation is very accurate when the feedback path is short. When the feedback path is long, disallowing oscillation and state-holding behavior is an optimistic assumption that makes the fault easier to diagnose. (Since the MMA theorem requires that potentially detecting vectors be ignored, disallowing potential detections gives us more information to use for diagnosis.)

4.1 MMA theorem interpretation

An ambiguity exists in the MMA theory as presented in the original paper. The theory consists of two main elements: a vector that detects a bridging fault will also detect an associated stuck-at fault, and therefore the signature of a bridging fault will be contained in its composite signature. The ambiguity arises because the concept of detection is described in terms of the test vector, but little or nothing is specified about the (error-carrying) circuit outputs.

As described in Section 3.3, the Byzantine Generals Problem may cause errors to appear on more or fewer circuit outputs than would otherwise be the case; whether more or fewer, the evidence presented by the application of one vector to the faulty circuit will not be identical to the expected output for any of the four single stuck-at faults associated with the two bridged nodes. Whether or not the faulty signature is contained in the composite signature is a question of the interpretation of the MMA term *contained.*

A narrow interpretation is that, for each vector, a faulty response is contained in the composite signature only if it is **indistinguishable from** the response of one of the four stuck-at faults; in other words, every output that is expected to evince an error for the stuck-at fault carries an error for the bridging fault. A broad interpretation is that the faulty response is contained if its affected outputs are **a subset of** the affected outputs of the four associated stuck-at faults. If the Byzantine Generals Problem were not a factor, the narrow interpretation would always be the superior choice: it strengthens the conditions for matching, thus reducing the number of matches in a diagnosis. However, the narrow interpretation coupled with the Byzantine Generals Problem will increase the number of incorrect diagnoses.

This paper consistently reports results using the broad interpretation of containment. After performing experiments using both interpretations on the original MMA technique and the improved technique, the results show that the strict interpretation results in, on average, 50% more incorrect diagnoses as with the broad interpretation. On the other hand, the average match sizes with the narrow interpretation are, on average, 60% of the broad-interpretation average match sizes.

The choice was made to use the broad interpretation for two reasons. First, a loss of precision in the matching process was considered acceptable to avoid misleading or incorrect diagnoses; as will subsequently be shown, other improvements made to the technique have improved the precision of the diagnoses greatly without significantly reducing their accuracy. Second, the narrow interpretation assumes that a bridging fault will behave, for a particular detecting vector, exactly or ostensibly as a stuck-at fault; as mentioned previously, this assumption has been shown to be unrealistic and largely unserviceable for successful bridging fault diagnosis.

4.2 Baseline results

This section reports the results from the original technique on the ISCAS-85 circuits. There are three important differences between these experiments and those reported in the original MMA paper. First, fault candidates have been limited to only realistic bridging faults, instead of all possible node pairs. Second, the behaviors diagnosed included the effects of the Byzantine Generals Problem, as described in Section 3.3. Third, all circuits reported are much larger than any circuit demonstrated in the original paper. Before performing the experiment presented in this section, the original MMA results were replicated on the ISCAS-85 circuits by setting all gate logic thresholds to the same value. As expected, no incorrect diagnoses occurred under these conditions.

Circuit	Total Trials	Average Matches	Percent Exact	Percent ≤ 10
C432	160	66.9	11.9%	29.4%
C499	279	71.1	19.4%	40.1%
C880	328	32.7	27.4%	60.7%
C1355	443	134.2	19.6%	43.8%
C1908	475	168.8	16.2%	32.6%
C2670	1371	232.6	12.6%	40.9%
C3540	1646	53.6	27.1%	55.9%
C5315	4043	102.6	29.3%	66.0%
C6288	2192	93.4	18.5%	33.8%
C7522	5379	248.6	17.0%	47.7%

Table 5: The original MMA technique: The total number of experimental trials, the average number of matches per diagnosis, percentage of diagnoses that are exact, and percentage of diagnoses of size 10 or smaller.

The diagnoses returned by the MMA technique are frequently unusable because of their size. Table 5 shows that the diagnoses returned by the MMA technique range from an average of 33 faults for the C880 to almost 250 faults for the C7552. The average number of faults is less than those reported by Millman, McCluskey, and Acken—even though their circuits were smaller—because the list of candidate faults is limited to realistic faults. Note also that no ordering of candidates is performed by the MMA technique: every fault in these large diagnoses is equally implicated, so no hint is given as to where to begin the physical search for the defect.

Table 5 also shows that the number of exact diagnoses is less than 30% for each circuit. Generally, around 50% of the diagnoses are of size ten or less and contain the correct fault. In order for the technique to be useful as a practical diagnostic tool, the percentage of the diagnoses that are small and contain the correct fault must be much larger than that afforded by the basic technique.

Circuit	Incorrect	Incomp	Mislead	Failed
C432	1.9%	1.9%	0%	0%
C499	7.9%	1.4%	0.4%	6.1%
C880	1.2%	0.6%	0%	0.6%
C1355	2.9%	0.7%	0%	2.3%
C1908	7.2%	0.4%	0.2%	6.5%
C2670	9.2%	2.4%	0.1%	6.6%
C3540	7.7%	1.8%	0.2%	5.7%
C5315	5.9%	1.6%	0.2%	4.1%
C6288	4.5%	1.0%	0.3%	3.2%
C7522	10.5%	3.3%	0.6%	6.5%

Table 6: The original MMA technique: The percentage of diagnoses that are incorrect: comprised of incomplete, misleading, and failed diagnoses.

Incorrect diagnoses can occur when the bridging faults to be diagnosed are affected by the Byzantine Generals Problem. Table 6 shows that an incorrect diagnosis occurs generally less than 10% of the time. An incorrect diagnosis can be a failed diagnosis, an incomplete diagnosis, or a misleading diagnosis; as shown, the incorrect diagnoses are dominated by failed diagnoses. Incomplete diagnoses are rare, and misleading diagnoses are almost non-existent. The domination of failed diagnoses in the incorrect diagnoses is an important feature of the MMA technique: when the technique does not provide the right answer, it rarely misleads the user.

5 Improved bridging fault diagnosis

While the original MMA technique is attractive because of its use of simple stuck-at signatures for diagnosing bridging faults, the previous section demonstrated some of the inadequacies of the original technique: large average diagnoses, unordered fault candidates, and a significant percentage of failed diagnoses. The goal of this research was to improve the MMA technique by addressing each of these issues, resulting ultimately in a technique that was not only relatively inexpensive to implement, but also precise, accurate, and able to compensate for failures resulting from the effects of the Byzantine Generals Problem. This section presents the improved technique, and discusses the several ways in which its diagnostic potential and precision have been enhanced over the original MMA technique.

5.1 Match restriction

A weakness of the MMA technique is that a faulty signature is likely to be contained in a large number of composite signatures. The larger a composite signature, the broader the range of potential matches, and the less likely it is to match only the faulty signature representing a bridge between the two candidate nodes. If unreasonable portions of the composite signature could be identified and removed, the result would be fewer matches per diagnosis and a commensurate increase in diagnostic precision.

The restriction employed here is to eliminate from a composite signature any entries that cannot be used to detect the described bridging fault. In order for a bridging fault to be detected, a test vector must stimulate opposite logic values on the two bridged nodes. Removing vectors that place identical values on the bridged nodes results in a composite signature that more precisely contains the possible behavior of the bridging fault.

Any vector in a composite signature that detects the same-valued stuck-at fault on both bridged nodes must stimulate the same value on both nodes; such a vector cannot detect the bridging fault and can be dropped from the composite signature. For example, a vector that detects both X stuck-at 0 and Y stuck-at 0 cannot detect X bridged to Y, and this vector can be removed from the composite signature for X bridged to Y. In terms of the notation introduced before, match restriction can be expressed as

$$\mathbf{U_{X@Y}} = \{\forall\langle o : v\rangle \mid (v \in \mathbf{S_{X0}} \wedge v \in \mathbf{S_{Y0}}) \vee (v \in \mathbf{S_{X1}} \wedge v \in \mathbf{S_{Y1}})\}, \tag{5}$$

where $\mathbf{U_{X@Y}}$ represents the set of restricted (unallowed) vectors for the bridging fault X bridged to Y. The composite signature under the improved technique is therefore

$$\mathbf{C_{X@Y}} = \mathbf{S_{X0}} \bigcup \mathbf{S_{X1}} \bigcup \mathbf{S_{Y0}} \bigcup \mathbf{S_{Y1}} - \mathbf{U_{X@Y}}. \tag{6}$$

Note that this method is not exhaustive—there are probably other vectors that place identical values on the bridged nodes—but this improvement requires no more information than that contained in the stuck-at signatures.

Exhaustive information, however, may be available, usually from a logic simulator or from the fault simulator used to generate the test set; a subsequent section presents the improvement in diagnostic precision that can be achieved using such information. The exhaustive form of match restriction can be represented as

$$\mathbf{U_{X@Y}} = \{\forall\langle o : v\rangle \mid \nu(\mathrm{X}, v) = \nu(\mathrm{Y}, v)\}. \tag{7}$$

A strength of match restriction, regardless of the source of node logic value information, is that it is not affected by the Byzantine Generals Problem—it can never increase the number of incorrect diagnoses.

5.2 Match requirement

While the match restriction of the previous section relied on identifying test vectors that *cannot* detect a particular bridging fault, the improvement presented in this section is based on vectors that *should* detect a bridging fault—namely, those vectors that place opposite logic values on the bridged nodes and detect single stuck-at faults on both of the bridged nodes. The second improvement to the MMA technique is based on identifying such vectors in the composite signatures, and then enforcing a *match requirement* on those vectors.

If during the construction of a composite signature a single vector is recognized as detecting both X stuck-at 0 and Y stuck-at 1 (or X stuck-at 1 and Y stuck-at 0), it is marked as a *required* vector. The set of required vectors, $\mathbf{R_{X@Y}}$, can be defined as

$$\mathbf{R_{X@Y}} = \{\forall\langle o : v\rangle \mid (v \in \mathbf{S_{X0}} \wedge v \in \mathbf{S_{Y1}}) \;\vee\; (v \in \mathbf{S_{X1}} \wedge v \in \mathbf{S_{Y0}})\}. \tag{8}$$

In order for the composite signature to match with an observed fault signature, the observed behavior must contain errors for all required vectors.

Figure 3 illustrates the composite signature of a fault candidate for node X bridged with node Y; it shows how the candidate is composed of the four single stuck-at behaviors for X and Y. When X stuck-at 0 and Y stuck-at 0 are both detected (or both stuck-at 1s are detected), the bridging fault cannot be stimulated, and those *restricted* vectors are removed from the fault candidate. When a particular vector detects both X stuck-at 0 and Y stuck-at 1 (or vice-versa), that vector should detect the bridging fault, and it is flagged as a *required* vector.

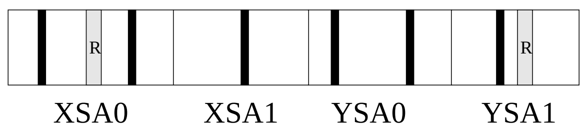

Figure 3: The composite signature of X bridged to Y with match restrictions (in black) and match requirements (labeled R)

Unlike match restriction, requiring matches can eliminate the correct match from a diagnosis. A vector may detect opposite stuck-at values and still fail to detect the bridging fault because the Byzantine Generals Problem could prevent fault propagation. In addition, if the bridge has a comparatively large resistance, certain vectors may not cause a propagatable error, which may also result in the elimination of the correct match from the diagnosis.

The example diagnoses from Tables 1 and 2 are revisited in Tables 7 and 8. These examples demonstrate the application of match restrictions and match requirements and the more precise diagnoses that result. The dictionary in the Table 7 is the same stuck-at fault dictionary shown in the example of Table 1. The dictionary in Table 8 results from the application of match restrictions and match requirements. For example, in the composite signature for fault candidate A@B, Vector 2 has been eliminated by match restriction, since it is found in the stuck-at signatures for both Asa0 and Bsa0. Also, Vector 4 has been marked as a required vector, indicated by the box, since it appears in the stuck-at signatures for both Asa0 and Bsa1. The result is that each of the example observed behaviors mentioned previously now matches only one candidate.

5.3 Match ranking

The original MMA technique did not order the elements of a diagnosis; a diagnosis simply consisted of an unranked list of matching candidate faults, with no expression of preference or likelihood assigned to the candidates. An overall ordering and expression of confidence for each candidate

Fault	Vector(s)
Asa0	2, 4
Asa1	1
Bsa0	2
Bsa1	4, 5
Csa0	4
Csa1	1, 3

Table 7: Stuck-at signatures

Fault	Vector(s)
A @ B	1, $\boxed{4}$, 5
B @ C	1, 2, 3, $\boxed{4}$, 5
C @ A	2, 3

Table 8: Composite signatures

would be useful for fault location, both to guide the physical search for defects and to give some indication of the quality of the diagnosis.

An additional important benefit can also result from ranking matches during diagnosis. As described in Section 3.3 and demonstrated experimentally in Section 4.2, the complications of bridging fault behavior can cause an MMA diagnosis to fail (not provide any matching candidate faults). By ranking all composite signatures, however, a failed diagnosis can be recovered by constructing a diagnosis out of the best, or highest-ranked, non-matching candidates.

The original MMA technique had a *strict* matching criterion: either a candidate contained the observed behavior, or it was eliminated from consideration. The improved technique, by assigning a measure of goodness to every candidate, can instead order the candidates and provide a diagnosis where the original technique would simply fail.

The sole expression of match goodness in the original MMA technique was that the observed faulty behavior be contained in a candidate composite signature. A more refined set of ordering criteria can be obtained by examining the elements of a typical match in greater detail.

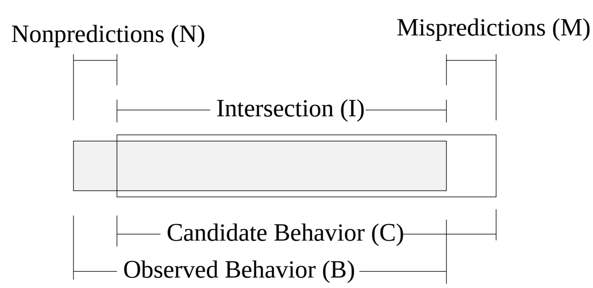

Figure 4: Overlapping of candidate behavior and observed behavior

Figure 4 shows the comparison between the observed behavior **B** (shaded) and a candidate fault **C** (unshaded). The observed output errors that are correctly predicted by the candidate are represented as set **I** (the *intersection*), the output errors that are predicted by the candidate but not observed are set **M** (the *mispredictions*—this is also known as the number of nodetects [18]), and the output errors that are observed but not predicted by the candidate are **N** (the *nonpredictions*).

The original MMA technique had a single, simple criterion for inclusion in a diagnosis motivated by the design of composite signatures. Composite signatures are deliberately inclusive—they predict many behaviors that are not expected to occur due to the presence of a bridging fault. However, the MMA theorem reported in Section 3 of this paper states unequivocally that the observed behavior must be contained in the correct composite signature. The behavior is deliberately over-predicted, so while a large amount of misprediction is anticipated, an unpredicted behavior is completely unexpected and so disqualifies a candidate. This is exactly the criterion used in the original MMA technique for inclusion of candidates in a diagnosis, which is restated here:

$$\mathrm{D} = \{\forall \mathrm{C_i} \mid \mathrm{B} \subseteq \mathrm{C_i}\}. \tag{9}$$

Another way to state this is that $|\mathbf{N_i}|$ must be 0 in order for candidate $\mathbf{C_i}$ to be included in the diagnosis.

Note that the improved technique implies additional criteria via set $\mathbf{R_i}$, the match requirements for candidate $\mathbf{C_i}$: the observed behavior must contain all of the match requirements of the candidate, or the candidate is eliminated. A strict-inclusion diagnosis in the improved technique could then be defined as

$$\mathrm{D} = \{\forall \mathrm{C_i} \mid (\mathrm{B} \subseteq \mathrm{C_i}) \wedge (\mathrm{R_i} \subseteq \mathrm{B})\}, \tag{10}$$

where $\mathbf{C_i}$ the improved composite signature (restricted vectors removed) defined in Section 5.1.

The strict matching described above expresses the important expectations about candidate faults as absolute criteria: either certain conditions are met, or the candidate is eliminated. The idea behind ranking candidates is to turn these severe accept-or-exclude criteria into a quantitative measure of relative match goodness.

The priorities for judging matches are the same as expressed in the previous section: the primary assertion is that the best candidates are the ones that contain the largest amount of the faulty behavior. Therefore, a candidate that contains a larger percentage of the observed behavior is considered superior to one that contains a smaller percentage; the primary quantitative measure of match goodness is the size of the intersection of the composite signature and observed behavior, or $|\mathbf{I}|$.

If this first criterion does not provide enough information to differentiate candidates, the application of match requirements establishes another expectation: the correct candidate will usually contain the required vectors. In order to formalize this expectation, first define $\mathcal{R}_i$, the percentage of predicted required vectors that are fulfilled by the candidate; if there are no required vectors, $\mathcal{R}_i$ is 1.

$$\mathcal{R}_\mathrm{i} = \begin{cases} 1 & \text{if } |\mathrm{R_i}| = 0 \\ \frac{|\mathrm{B} \cap \mathrm{R_i}|}{|\mathrm{R_i}|} & \text{otherwise} \end{cases}$$

Additionally, there exists a third parameter to judge the quality of an individual match: the amount of misprediction. By removing restricted vectors, the expected size of $\mathbf{M}$ has been minimized. Given the simplifying assumption that all such mispredictions are equally likely, if two candidates have the same size intersection with the observed behavior, and contain the same percentage of required vectors, but one candidate has a much larger $\mathbf{M}$ than the other, this candidate is considered a less-likely description for the faulty behavior.

The total ranking, then, is a lexicographic ordering in which the nonprediction index has priority, followed by the prediction of the required vectors (used to break nonprediction index ties), and finally the misprediction value (used to break ties of the first two metrics). The ranking is expressed as

$$\forall \mathrm{C_i}\ (|\mathrm{I_i}| \geq |\mathrm{I_{i+1}}|), \tag{11}$$

$$\forall \mathrm{C_i}\ (|\mathrm{I_i}| = |\mathrm{I_{i+1}}|) \longrightarrow (\mathcal{R}_\mathrm{i} \geq \mathcal{R}_\mathrm{i+1}), \tag{12}$$

$$\forall \mathrm{C_i}\ (|\mathrm{I_i}| = |\mathrm{I_{i+1}}|) \land (\mathcal{R}_\mathrm{i} = \mathcal{R}_\mathrm{i+1}) \longrightarrow (|\mathrm{M_i}| \leq |\mathrm{M_{i+1}}|). \tag{13}$$

Figure 5 shows four candidate faults compared with the observed behavior and their resulting ranking. $\mathbf{C_1}$, $\mathbf{C_2}$, and $\mathbf{C_3}$ are ranked higher than $\mathbf{C_4}$ because they contain more of the observed behavior, as specified in Equation (5.7); $\mathbf{C_1}$ and $\mathbf{C_2}$ are ranked higher than $\mathbf{C_3}$ because they contain a higher percentage of required vectors, as specified in Equation (5.8); and $\mathbf{C_1}$ is ranked higher than $\mathbf{C_2}$ because it contains less misprediction, as specified in Equation (5.9).

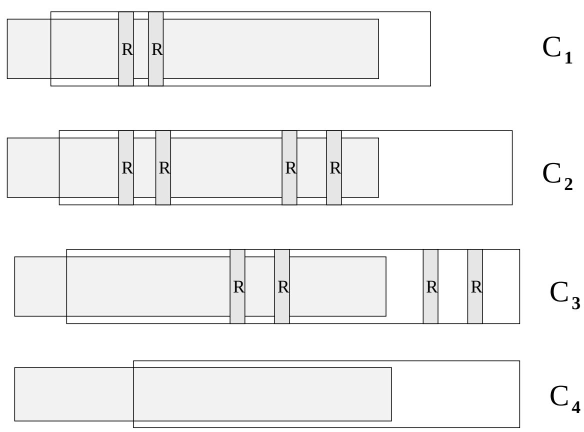

Figure 5: Four candidate faults (unshaded) compared with the same observed behavior (shaded), and their corresponding rankings: $\mathrm{C_1}$ is the highest-ranked candidate, and $\mathrm{C_4}$ is the lowest.

Given that the composite signature candidates are now ranked, it is a simple matter to relax the strict criteria for inclusion in a diagnosis, resulting in a policy that will recover diagnoses that would otherwise fail.

A standard technique used by many diagnostic algorithms [36, 18] is to first select an acceptable diagnosis size, d, and then construct a diagnosis of the d highest-ranked candidates. Combining this simple limit with the rankings defined in the previous section, the final rankings and criteria for inclusion in a diagnosis in the improved technique are therefore the previous Equations 11 through 13 with the follwing addition:

$$\mathrm{D} = \bigcup_{\mathrm{i}=1}^{\mathrm{d}} \{\mathrm{C_i}\}. \tag{14}$$

In the results reported in Section 5.4, the ranking of the correct match is reported as an evaluation of the match ranking technique. In addition, results are given that show the number of diagnoses that would fail under the strict matching criteria of original MMA, and the number that are successfully recovered (the correct match is found in the top d candidates) by the improved technique. The results demonstrate that the ranking system does indeed reflect the quality of the matches in a diagnosis, since the correct match is almost always ranked among the top three candidates selected.

5.4 Improved results

The improved technique uses a different set of criteria to construct a diagnosis (as described in Section 5.3, making the direct comparison of baseline (MMA) and improved results difficult. To allow this comparison, the experimental results for the new technique have been processed and presented in the same form used in Section 4.2 for results from the original technique: The results given in Tables 9 through 12 show what results if strict matching is used in the improved technique[7]. The technique is not prohibitively expensive in computation time: the 5,379 experiments run on the C7552 took an average of 18.9 seconds per diagnosis on an Alphastation 250 4/266.

Circuit	Total Trials	Average Matches	Percent Exact	Percent ≤ 10
C432	160	3.4	48.8%	96.8%
C499	279	1.3	74.9%	90.3%
C880	328	1.3	82.0%	93.9%
C1355	443	1.6	75.6%	95.5%
C1908	475	7.5	54.3%	84.9%
C2670	1371	10.4	53.6%	84.8%
C3540	1646	3.4	62.3%	85.6%
C5315	4043	4.5	72.7%	83.1%
C6288	2192	1.0	86.7%	90.0%
C7522	5379	10.5	55.2%	72.1%

Table 9: The improved technique under strict matching, using only single stuck-at information for match restriction: The total number of experimental trials, average number of matches per diagnosis, percentage of diagnoses that are exact, and percentage of diagnoses that have ten or fewer matches and contain the correct match.

The diagnoses returned by the improved technique are a substantial improvement over the original technique. Table 9 shows that for each circuit, the size of the average diagnosis is less than one twelfth of its previous value; in some cases the average diagnosis is ninety times smaller than before (compare to Table 5). For five of the benchmark circuits, the correct match is part of a small diagnosis (size ten or less) more than ninety percent of the time.

Table 10 shows that the number of diagnoses that do not contain the correct match has increased by a few percent (compare to Table 6) because of the interaction between the match requirement

[7]The results for the other two fault samples previously mentioned in Footnote 6 were statistically indistinguishable from the reported results.

Circuit	Incorrect	Incomp	Mislead	Failed
C432	2.5%	0.6%	0%	1.9%
C499	9.7%	0%	0%	9.7%
C880	6.1%	1.2%	0.6%	4.3%
C1355	3.4%	0%	0%	3.4%
C1908	8.0%	1.3%	0%	6.7%
C2670	11.9%	1.1%	0.7%	10.1%
C3540	10.8%	0.5%	0.4%	9.9%
C5315	12.2%	0.4%	0.2%	11.5%
C6288	9.8%	0%	0%	9.8%
C7522	17.1%	0.9%	0.7%	15.4%

Table 10: The improved technique under strict matching, using only single stuck-at information for match restriction: The percentage of diagnoses that are incorrect, comprised of incomplete, misleading, and failed diagnoses.

technique and the Byzantine Generals Problem, but the vast majority of the increase comes from failed diagnoses and not misleading diagnoses. The increased number of failures is more than offset by the increase in usable diagnoses; ultimately, failures can be eliminated by relaxing the matching criteria, results for which will be presented shortly.

As shown in Tables 11 and 12, the results improve even further if information is included about internal node values from logic simulation when establishing match restrictions. The primary parameter of interest, the size of the average diagnosis, approaches the ideal of 1.0 for most of the circuits.

Circuit	Total Trials	Average Matches	Percent Exact	Percent ≤ 10
C432	160	1.8	76.3%	93.8%
C499	279	1.1	84.2%	90.3%
C880	328	1.1	89.3%	93.3%
C1355	443	1.3	86.5%	95.5%
C1908	475	3.5	67.4%	85.1%
C2670	1371	7.0	62.4%	78.3%
C3540	1646	1.3	79.4%	87.9%
C5315	4043	1.1	82.2%	87.5%
C6288	2192	1.0	87.3%	90.2%
C7522	5379	2.4	67.4%	79.7%

Table 11: The improved technique under strict matching, using logic simulation for match restriction: The total number of experimental trials, the average number of matches per diagnosis, percentage of diagnoses that are exact, and percentage of diagnoses that have ten or fewer matches and contain the correct match.

As mentioned, ranking the matches and relaxing the matching criteria can both indicate the quality of individual matches and eliminate failed diagnoses. Table 13 shows these two effects. The

Circuit	Incorrect	Incomp	Mislead	Failed
C432	2.5%	0.6%	0%	1.9%
C499	9.7%	0%	0%	9.7%
C880	6.1%	0.3%	0.6%	5.2%
C1355	3.4%	0%	0%	3.4%
C1908	8.9%	0.4%	0.4%	8.0%
C2670	12.3%	0.5%	0.7%	11.0%
C3540	11.2%	0.1%	0.1%	10.9%
C5315	12.0%	0%	0%	12.0%
C6288	9.8%	0%	0%	9.8%
C7522	16.6%	0.3%	0.4%	15.9%

Table 12: The improved technique under strict matching, using logic simulation for match restriction: The percentage of diagnoses that are incorrect, comprised of incomplete, misleading, and failed diagnoses.

first and third data columns report the average position of the correct match for diagnoses that did not fail the strict criteria, showing that ranking the matches can further improve the results above by highly-ranking the correct (desired) match. The second and fourth data columns show the effects of relaxing the matching criteria on the failed diagnoses. A failed diagnosis under strict matching criteria is considered recovered if the correct match is included in the d (in this case ten) candidates of the relaxed-criteria diagnosis; the **Recovered** sub-column reports the success rate of this recovery process. The **Ave Pos** sub-column reports the average rank of the correct match in these recovered diagnoses; again, the ranking is quite successful, ranking the correct match in the top two or three candidates.

Circuit	No logic simulation			With logic simulation		
	Non-Failed	Failed		Non-Failed	Failed	
	Ave Pos	Recovered	Ave Pos	Ave Pos	Recovered	Ave Pos
C432	1.9	2/3	1.2	4.0	3/3	2.3
C499	1.0	27/27	1.0	1.9	27/27	1.5
C880	1.1	13/14	1.0	1.2	17/17	1.5
C1355	1.4	15/15	1.2	1.1	15/15	1.1
C1908	3.1	28/32	1.4	1.5	32/35	1.3
C2670	2.3	136/139	1.4	1.2	142/144	1.2
C3540	2.2	157/163	1.1	1.6	173/175	1.4
C5315	3.2	440/466	1.0	1.4	474/482	1.1
C6288	1.0	210/214	1.0	1.3	211/215	1.3
C7522	6.0	732/826	1.6	1.4	806/856	1.4

Table 13: The effects of match ranking, and relaxing the match criteria for failed diagnoses: The average position of the correct match in non-failed (strict-criteria) diagnoses, the number of failed diagnoses recovered and the average position of the correct match in the recovered diagnoses. **No Sim** and **Sim** indicate that match restriction was applied without and with logic simulation, respectively.

6 Comparison to another approach

In order to compare the improved results to standard diagnosis methods, this section presents an experiment modeled after the Teradyne fault diagnosis system [37]. The *fault ordering* method penalizes candidate stuck-at faults for each predicted failure that did not actually occur and for each failure that occurs without being predicted by the candidate. This procedure produces a ranked list of stuck-at faults. Given this ranking, if any of the four stuck-at faults associated with the two bridged nodes appears among the ten highest-ranked faults, the diagnosis is considered a success; otherwise it is counted as a misleading diagnosis. The same observed faulty behaviors used to evaluate the original MMA technique and the improved technique were diagnosed for this technique: the top 10% most likely bridging faults.

Circuit	Avg N1 position	Avg N2 position	Misleading Diagnoses
C432	2.1	48.2	0.6%
C499	2.9	44.8	5.4%
C880	1.6	34.6	0%
C1355	5.0	92.8	9.0%
C1908	1.9	77.0	1.3%
C2670	3.5	94.7	6.6%
C3540	1.6	102.3	0.6%
C5315	1.5	52.8	0.8%
C6288	5.8	228.0	14.4%
C7522	2.2	107.8	3.0%

Table 14: Fault ordering: average position of the first node, average position of the second node, and the percentage of misleading diagnoses

Table 14 shows the results of fault ordering. The average position of the first node is in the first ten faults, but the average position of the second node is far behind the first node. The number of diagnoses where neither node appears in the top ten nodes is substantially larger than the number of misleading diagnoses for either the original or improved MMA techniques.

The improved technique is better than the fault ordering technique in two respects. First, the improved technique provides the exact two nodes of interest in a set of ten or fewer most of the time. Providing the exact pair is superior to providing individual candidate stuck-at faults: every candidate stuck-at fault represents one node that could be involved with many potential realistic defects. Second, while the improved MMA technique may produce an incorrect diagnosis, an unrecovered incorrect diagnosis occurs less than 4% of the time. All fault ordering diagnoses appear to be the same; there is no way to distinguish a misleading diagnosis from good diagnosis.

7 Conclusions

This paper has presented the elements of fault diagnosis in general, and the details of bridging fault diagnosis in particular. It has presented the implementation and analysis of a specific technique for diagnosing bridging faults with inexpensively-obtained stuck-at signatures, and has shown that this technique is unusable for even small CMOS circuits because of the large size of the average diagnosis. Also, it has demonstrated that the Byzantine Generals Problem causes incorrect diagnoses, although the number of misleading diagnoses is very low.

This paper presents an improved technique for diagnosing bridging faults with stuck-at signatures, with several important changes to the original technique. First, only those faults determined to be realistic through inductive fault analysis are considered as candidates. The number of realistic faults is much smaller than the number of all theoretically possible bridges, not only improving the precision of the diagnoses, but making the technique feasible for much larger circuits. Second, match restrictions and match requirements are imposed during matching in order to minimize diagnosis size. Finally, match ranking is applied and the matching criteria relaxed to further increase the effective precision and to increase the number of correct diagnoses. Using all described improvements, the reported experiments show that at least 90% of the time the correct match is found in a diagnosis of size ten or less, a significant indication of diagnostic effectiveness.

References

[1] M. Abramovici, M. A. Breuer, and A. D. Friedman. *Digital Systems Testing and Testable Design*. Computer Science Press, 1990.

[2] J. M. Acken. *Deriving Accurate Fault Models*. PhD thesis, Stanford University, Department of Electrical Engineering, September 1988.

[3] J. M. Acken and S. D. Millman. Accurate modeling and simulation of bridging faults. *Proceedings of the Custom Integrated Circuits Conference*, pages 17.4.1–17.4.4, 1991.

[4] J. M. Acken and S. D. Millman. Fault model evolution for diagnosis: Accuracy vs precision. *Proceedings of the Custom Integrated Circuits Conference*, 1992.

[5] R. C. Aitken. Fault locationwith current monitoring. In *Proceedings of International Test Conference*, pages 623–632. IEEE, 1991.

[6] R. C. Aitken. A comparison of defect models for fault location with iddq measurements. In *Proceedings of International Test Conference*, pages 778–787. IEEE, 1992.

[7] R. C. Aitken. Finding defects with fault models. In *Proceedings of International Test Conference*, pages 498–505. IEEE, 1995.

[8] R. C. Aitken and P. C. Maxwell. Better models or better algorithms? on techniques to improve fault diagnosis. *Hewlett-Packard Journal*, February 1995.

[9] M. A. Breuer and A. D. Friedman. *Diagnosis and Reliable Design of Digital Systems*. Computer Science Press, 1976.

[10] F. Brglez and H. Fujiwara. A neutral netlist of 10 combinational benchmark circuits and a target translator in fortran. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, 1985.

[11] D. Burns. Locating high resistance shorts in CMOS circuits by analyzing supply current measurement vectors. In *International Symposium for Testing and Failure Analysis*, pages 231–237, November 1989.

[12] P. Camurati, A. Lioy, P. Prinetto, and M. S. Reorda. Diagnosis oriented test pattern generation. In *Proc. European Design Automation Conf.*, pages 470–474, 1990.

[13] S. Chakravarty and Y. Gong. An algorithm for diagnosing two-line bridging faults in combinational circuits. *Proceedings of Design Automation Conference*, pages 520–524, 1993.

[14] S. Chakravarty and M. Liu. $\mathrm{i}_{\mathrm{ddq}}$ measurement based diagnosis of bridging faults. *Journal of Electronic Testing: Theory and Applications*, January 1993.

[15] B. Chess. Diagnostic test pattern generation and the creation of small fault dictionaries. Master's thesis, University of California, Santa Cruz, June 1995.

[16] B. Chess and T. Larrabee. Bridge fault simulation strategies for CMOS integrated circuits. In *Proceedings of Design Automation Conference*, pages 458–462, 1993.

[17] B. Chess, T. Larrabee, and C. Roth. On evaluating competing bridge fault models for CMOS ICs. In *Proceedings of the 1994 VLSI Test Symposium*, pages 446–451. IEEE, 1994.

[18] K. De and A. Gunda. Failure analysis for full-scan circuits. In *Proceedings of International Test Conference*, pages 636–645. IEEE, 1995.

[19] D. Feltham and W. Maly. Physically realistic fault models for analog CMOS neural networks. *IEEE Journal of Solid-State Circuits*, 26(9):1223–1229, September 1991.

[20] F. J. Ferguson and T. Larrabee. Test pattern generation for realistic bridge faults in CMOS ICs. In *Proceedings of International Test Conference*, pages 492–499. IEEE, 1991.

[21] A. Gattiker and W. Maly. Current signatures. In *Proceedings of the 1996 VLSI Test Symposium*, pages 112–117. IEEE, 1996.

[22] A. Gattiker and W. Maly. Current signatures: Application. In *Proceedings of International Test Conference*, pages 156–165. IEEE, 1997.

[23] T. Gruning, H. Koopmeiners, and U. Mahlstedt. DIATEST: A fast diagnostic test pattern generator for combinational circuits. In *Proceedings of International Conference on Computer-Aided Design*, pages 194–197. IEEE, 1991.

[24] A. Jee and F. J. Ferguson. Carafe: A software tool for failure analysis. In *Proceedings of International Symposium for Testing and Fault Analysis*, pages 143–149, 1993.

[25] A. Jee and F. J. Ferguson. Carafe: An inductive fault analysis tool for CMOS VLSI circuits. In *Proceedings of the IEEE VLSI Test Symposium*, pages 92–98, 1993.

[26] J.Jacob and N. N. Biswas. Gtbd faults and lower bounds on multiple fault coverage of single fault single fault test sets. In *Proceedings of International Test Conference*, pages 849–855. IEEE, 1986.

[27] K. Kubiak, S. Parkes, W. K. Fuchs, and R. Saleh. Exacts evaluation of diagnostic test resolution. In *Proceedings of Design Automation Conference*, pages 347–352. IEEE, 1992.

[28] R. P. Kunda. Fault location in full-scan designs. In *International Symposium for Testing and Failure Analysis*, pages 121–127. IEEE, 1993.

[29] M. Pease L. Lamport, R. Shostak. The byzantine Generals Problem. Technical report, Comp. Sci. Lab, SRI International, March 1980.

[30] T. Larrabee. Test pattern generation using Boolean satisfiability. *IEEE Transactions on Computer-Aided Design*, pages 4–15, January 1992.

[31] P. Maxwell and R. Aitken. Biased voting: a method for simulating CMOS bridging faults in the presence of variable gate logic thresholds. In *Proceedings of International Test Conference*, pages 63–72. IEEE, 1993.

[32] K. C. Y. Mei. Bridging and stuck-at faults. *IEEE Transactions on Computers*, C-23(7):720–727, July 1974.

[33] S. D. Millman, E. J. McCluskey, and J. M. Acken. Diagnosising CMOS bridging faults with stuck-at fault dictionaries. In *Proceedings of International Test Conference*, pages 860–870. IEEE, 1990.

[34] M. M. Ervin-Willis R. W. Allen and R. E. Tullose. Dora: Cad interface to automatic diagnostics. In *Proceedings of Design Automation Conference*, pages 559–563. IEEE, 1982.

[35] J. Rajski and H. Cox. A method of test generation and fault diagnosis in very large combinational circuits. In *Proceedings of International Test Conference*, pages 932–943. IEEE, 1987.

[36] V. Ratford and P. Keating. Integrating guided probe and fault dictionary: an enhanced diagnostic approach. In *Proceedings of International Test Conference*, pages 304–311. IEEE, 1986.

[37] J. Richman and K. R. Bowden. The modern fault dictionary. In *Proceedings of International Test Conference*, pages 696–702. IEEE, 1985.

[38] E. Rudnick, W. K. Fuchs, and J. Patel. Diagnositic fault simulation of sequential circuits. In *Proceedings of International Test Conference*, pages 178–186. IEEE, 1992.

[39] P.G. Ryan, W.K. Fuchs, and I. Pomeranz. Fault dictionary compression and equivalence class computation for sequential circuits. In *Proceedings of International Conference on Computer-Aided Design*, pages 508–511, 1993.

[40] J. P. Shen, W. Maly, and F. J. Ferguson. Inductive fault analysis of MOS integrated circuits. *IEEE Design and Test of Computers*, 2(6):13–26, December 1985.

[41] J. W. Sheppard and W. R. Simpson. *System test and diagnosis*. Kluwer, 1994.

[42] J. W. Sheppard and W. R. Simpson. Improving the accuracy of diagnostics provided by fault dictionaries. In *Proceedings of the 14th VLSI Test Symposium*, pages 180–185. IEEE, 1996.

[43] C. Thibeault. A novel probabilistic approach for IC diagnosis based on differential quiescent current signatures. In *Proceedings of the 1997 VLSI Test Symposium*, pages 80–85. IEEE, 1997.

[44] J. Waicukauski and E. Lindbloom. Failure diagnosis of structured vlsi. *IEEE Design and Test of Computers*, pages 49–60, August 1989.