On Test Syndrome Merging for Reasoning-Based Board-Level Functional Fault Diagnosis*

Zelong Sun†, Li Jiang‡, Qiang Xu†, Zhaobo Zhang§, Zhiyuan Wang§, Xinli Gu§

†Department of CS&E, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
‡Deptartment of CS&E, Shanghai Jiaotong University, Shanghai, China
§Huawei Technologies, Inc., Santa Clara, CA

ABSTRACT

Machine learning algorithms are advocated for automated diagnosis of board-level functional failures due to the extreme complexity of the problem. Such reasoning-based solutions, however, remain ineffective at the early stage of the product cycle, simply because there are insufficient historical data for training the diagnostic system that has a large number of test syndromes. In this paper, we present a novel test syndrome merging methodology to tackle this problem. That is, by leveraging the domain knowledge of the diagnostic tests and the board structural information, we adaptively reduce the feature size of the diagnostic system by selectively merging test syndromes such that it can effectively utilize the available training cases. Experimental results demonstrate the effectiveness of the proposed solution.

1. INTRODUCTION

Today's complicated electrical system consists of many printed circuit boards and each board contains tens of even hundreds of integrated circuits (ICs). No doubt to say, VLSI testing is a critical step to ensure the quality and reliability of the system. Due to imperfect defect models and limited testing time, however, subtle defects in ICs may pass manufacturing tests and result in defective chips being integrated into the system. Eventually, these defects will manifest themselves as functional failures at the board- and system-level [1]. Consequently, board-level test and diagnosis is an essential step to prevent shipping problematic products to the customers.

In board-level diagnosis, usually debug technicians from contract manufacturers run various functional diagnostic tests developed by the system vendor and try to identify the root-cause of the board failure based on syndromes produced by these tests. This, however, is an extremely challenging task, especially considering the fact that debug technicians typically have limited knowledge about the board under diagnosis (BUD) and the associated diagnostic tests [2].

In order to improve diagnosis accuracy, a number of automated diagnostic solutions have been proposed in the literature. Generally speaking, they can be categorized into two types. In the first type, based on the comprehensive understanding of the BUD and its diagnostic tests, experts from system vendor define rules [3] to guide the debug technicians, or model the underlying system directly for diagnosis [4]. With the increasing complexity of electrical systems, however, it is rather difficult to build very effective diagnostic rules and/or models, rendering relatively poor diagnosis accuracy. Instead of relying on expertise knowledge, the second approach resorts to machine learning and data analysis tools (e.g., support-vector machine (SVM) and artificial neural networks) to train a reasoning engine from the historical diagnostic data for diagnosis [5]. While promising, such reasoning-based solutions also have some limitations. On the one hand, insufficient historical data may lead to the so-called "overfitting" problem[6], which occurs when there are too many parameters (i.e., test syndromes) relative to the number of observations (i.e., the available historical data) during training. On the other hand, there may exist redundant or irrelevant test syndromes, causing ambiguous root-cause identification.

To mitigate the above issues in reasoning-based diagnostic solutions, Sun *et al.* [2] proposed a metric, namely *isolation capability*, to evaluate the quality of diagnostic tests, leveraging the structural information of the BUD and simple behavioral description of the diagnostic tests. They further utilized this metric to select the most effective test syndromes for adaptive diagnosis. Ye *et al.* [7] presented another evaluation metric based on information-theoretic analysis and used it to select those test syndromes with high discriminative ability. With effective test syndrome selection for feature reduction in machine learning algorithms [8], the above methods facilitate to improve diagnosis accuracy accordingly. At the same time, however, these methods may lose some useful information with certain test syndromes being excluded from the reasoning-based diagnostic engine.

In other words, while there is a need to reduce the number of test syndromes to resolve the "overfitting problem", especially at the early stage of the product cycle, at the same time, it is preferable to preserve useful information in the given test syndromes whenever possible. Consequently, in this work, we propose a new reasoning-based diagnostic framework, which relies on effective test syndrome merging instead of test syndrome selection for feature reduction. The proposed test syndrome merging algorithm takes advantage of the statistical history of the training data and domain knowledge about the BUD. It is also adaptive, i.e., the merging results vary with historical diagnosis data at different product cycle for more effective diagnosis. Experimental results demonstrate that the proposed solution is able to improve diagnosis accuracy for more than 10% for a complicated industrial board.

The remainder of this paper is organized as follows. Section 2 introduces existing diagnosis techniques for board-level functional failures and motivates this work. In Section 3, we propose the concept of test merging and details the test merging algorithm. Section 5 demonstrate the proposed diagnosis flow. Experimental results are then presented in Section 6. Finally, Section 7 concludes this work.

2. PRELIMINARIES

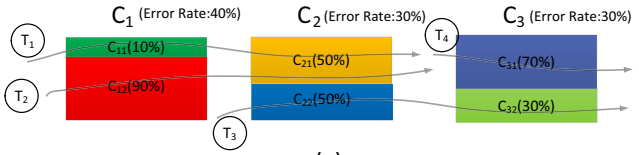
In this section, we first introduce reasoning-based diagnostic solutions and describe the associated challenges. Next, we present the motivations for the proposed test syndrome merging technique. We hereby define some terms for later use.

Ambiguity: the ratio of the faults that lead to make a wrong prediction among all possible faults in the board.

Predictable: a fault is predictable if the diagnosis database has at least one training case, in which the syndrome is identical to what the fault generates

*The work of Z. Sun, L. Jiang and Q. Xu was supported in part by a research grant from Huawei Technologies, Inc.

978-1-4799-7792-5/15/$31.00 ©2015 IEEE



(a)

Component	C_1		C_2		C_3	
Sub-component	C_{11}	C_{12}	C_{21}	C_{22}	C_{31}	C_{32}
Syndrome	1000	0100	1100	0010	0001	0010

(b)

Figure 1: Example board, tests and the corresponding test syndromes.

Predictability: the ratio of predictable faults among all possible faults in the board.

Error rate: the probability that a subcomponent/component is faulty when its parent component/board is faulty.

2.1 Reasoning-based Diagnosis

Reasoning-based diagnosis methods are emerging recently, because the sophisticated knowledge of mapping the syndrome to the root-cause can be incrementally built and maintained like a "black-box" with machine learning and data analysis tools, such as neural networks [5] and support-vector machine [9]. Trained by the historical diagnosis data, the resulting statistical model has the implicit capability to make the right diagnosis decisions even when the new syndromes have yet to appear in the historical database.

In a diagnosis system, a test syndrome composes of the pass/fail information of all diagnostic tests, typically in the range of several hundreds. A failed board with root-cause component C_i passes two diagnostic tests T_1 and T_3, but fails another two tests, T_2 and T_4 generate a test syndrome $< T_1 = 0, T_2 = 1, T_3 = 0, T_4 = 1 >$ (in short for $< 0, 1, 0, 1 >$). Fig. 1(a) illustrates a synthetic board composed of three components C_1, C_2, C_3, each of which contains two subcomponents. The arrows show the paths of the corresponding diagnostic tests. Fig. 1(b) shows a table with ideal syndromes for different root-causes, by assuming the diagnostic test has 100% coverage on the subcomponent it passes through. A successfully-diagnosed board provides a training case: a pair of syndrome and the root-cause component. Suppose in time t_1 of the product cycle, the diagnosis database has three training cases (see Fig 2). The statistical model is then trained by tuning the weight associated with each test for characterizing the root-cause candidates. To illustrate the reasoning-based diagnostic process, we employ a "one against all" linear SVM model in [9], wherein each component has a weight vector for all the tests. Given the above three training cases, for example, the weight vector associated with C_1 is tuned to be $< 66.7, 0.0, -33.3, -33.3 >$ (see the other two weight vectors for C_2 and C_3 in Fig. 2). When a new board is ready for diagnosis, a new syndrome vector is generated after testing. A decision function, which is vector dot product between the new syndrome vector and the weight vector associated with each component, is used to calculate the score. The component with highest score is considered as the predicted root-cause[9].

Modern electrical system requires hundreds of tests to fulfill the diagnosis requirement. Consequently, the weight/syndrome vector in the statistical model should contain hundreds of weights/tests. To train such a model, we need at least thousands of training cases that far surpass what the product line can afford. The lack of successful diagnosed board, especially in the early ramp-up phase, thus causes the so-called *overfitting* problem [6].

2.2 Overfitting Problem

In machine learning, *overfitting* is a typical scenario where a statistical model is too biased for the training cases and loses generality for future data. In the board diagnosis domain, the dimension of the test syndromes(hundreds to thousands) is too large relative to the number of successfully-diagnosed boards at the product ramp-up stage, which results in significant bias to outliners for the diagnosis prediction. Consider the example in Fig. 1, the statistical model has high confidence to diagnose any faults in C_{11}, C_{22} and C_{31}(as shown in Fig. 2). Suppose a failed board with root-cause C_{12} comes to the diagnosis system in $t_2 > t_1$, we only observe the test syndrome $< 0, 1, 0, 0 >$. As no matched syndrome exists in the diagnosis database, the statistical model has very low confidence to map it to C_1, leading to ambiguous diagnosis. According to the weight vector shown in Fig. 2, the decision function obtains $< 0, 1, 0, 0 > \cdot < 66.7, 0.00, -33.3, -33.3 >^T = 0$. In the example, we assume the *error rates* of the three components as 40%,30% and 30% respectively. With this, we can calculate the predictability of all the components on the whole board, which is the summation of diagnosable area weighted by the error rates $10\% \times 40\% + 50\% \times 30\% + 70\% \times 30\% = 40\%$. The reason for such ambiguity is that, for C_1, the statistical model can only predict faults in C_{11} (only 10% chance among all possible faults in C_1) rather than those in C_{12} based on existing training cases, rendering an "overfitted" (biased) model. Therefore, how to improve the generality of the statistical model for diagnosis with limited historical data is an interesting problem to solve.

2.3 Test Syndrome Selection

To mitigate the overfitting problem, an effective way is to reduce the feature size and hence simplify the statistical model, by selecting a subset of diagnostic test syndromes with maximum diagnosability.

Ye et al. [7] employ the typical information-theoretic analysis metric (i.e., mutual information) to form a test syndrome set with maximum relevance. Unfortunately, the metric itself depends on the historical data, and thus it cannot guarantee a general statistical property during product ramp-up stage. Sun et al.[2] propose a metric *isolation capability*(ICap), which evaluates the capability of the diagnostic test syndrome to discriminate each component. The metric is determined by predetermined information, such as the test paths, test coverage and the defect-per-million of each component, and thus is less affected by the historical data. Their method can adaptively reduce the syndrome's dimension according to the root-cause candidates in an iterative manner, and preserve the highest isolation capability, rendering a high successful diagnosis rate even when the training database is small. However, without explicitly considering the fact that the training cases increase in later product cycle, this method fails to further enhance the diagnosis accuracy in later stage of product cycle.

The underlying philosophy of test syndrome selection is to strike a balance between the amount of information contained in the test syndromes and the dimension of test syndrome. No matter how subtle the metric is, the information loss due to test selection contributes noticeable penalty for diagnosis. In the same example, we employ max-relevance[10]– a widely-accepted feature selection technique–to select three tests, T_1, T_3 and T_4 because they are most statistically relevant to C_1, C_2 and C_3 respectively. After T_2 is removed, the model has no ability to predict the fault conveyed by it. When a new syndrome $< 0, 1, 0, 0 >$ comes , we can calculate the score for C_1 using the new decision function, i.e., $< 0, 0, 0 > \cdot < 66.7, -33.3, -33.3 >^T = 0$, which is unable to make the right prediction. Motivated by above, in this work, we propose a novel feature reduction method, by merging instead of removing diagnostic test syndromes.

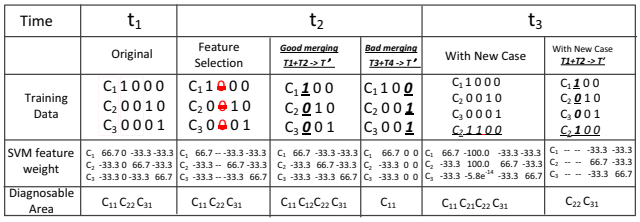


Figure 2: A motivational example. $t_1 < t_2 < t_3$ **are the time of product cycle.**

3. TEST SYNDROME MERGING

In this section, we first define the concept of test syndrome merging. Next, an intuitive example is presented to show i) the impacts of test syndrome merging for diagnosis and ii) the importance of an effective and efficient merging methodology. At last, we formulate the problem on how to conduct an effective test syndrome merging.

3.1 Definition

Test Syndrome Merging: Given a set of tests $\{T_1, T_2, ..., T_m\}$ and their paths $\{P(T_1), P(T_2), ..., , P(T_m)\}$, a *test merging* for these tests is defined as an action such that the merged new test T', with path $P(T')=\bigcup_{1\leq i\leq m} P(T_i)$ and the syndrome $S(T')=\bigcup_{i=1}^{n} S(T_i)$. Intuitively, the merged test can be treated as running a batch of individual test programs respectively, and therefore any failed (original) test indicates the failing of the merged test. In terms of test coverage, the merged test covers all the area that is covered by its contained tests.

3.2 An Intuitive Example

When the training cases are not enough, neither T_1 nor T_2 can solely convey the fault information of C_1. Fortunately, if we notice the structural relationship between T_1 and T_2 as well as the current training data, we can merge the two test syndromes into T'. And one of the training cases after merging becomes $\{C_1, < T' = 1, T_3 = 0, T_4 = 0 >\}$, which can correctly match to the observed syndrome after merging $< T' = 1, T_3 = 0, T_4 = 0 >$. The statistical model now is able to make the right prediction (see "good merging" column in Fig 2, since the score of C_1 is $1 \times 66.7 + 0 \times (-33.3) + 0 \times (-33.3) = 66.7$, which is higher compared to C_2 and C_3 whose scores are both 0. The predictability is improved to be $(10\% + 90\%) \times 40\% + 50\% \times 30\% + 70\% \times 30\% = 76\%$. The enhanced predictability is the key to mitigating the overfitting problem, by making the syndrome more informative. Because, with the help of the domain knowledge of board structure and test behavior, test syndrome merging endows the statistical model with extra predictability on potential root-causes absent in the diagnosis database.

However, the test syndrome merging, on the other hand, can result in side-effects (Ambiguity). Suppose we merge T_3 and T_4 in order to predict C_{32} from the training cases. The syndromes for root-cause C_2 and C_3 in diagnosis database become the same $< T_1 = 0, T_2 = 0, T' = 1 >$ as shown in "bad merging" column Fig. 2). The model then becomes ambiguous and has low confidence (the weights of the model shown in Fig. 2) to discriminate between C_2 and C_3. Therefore, a subtle merging algorithm should consider above side-effect and choose the suitable test syndromes for merging based on current diagnosis database and the structure information.

As more successfully-diagnosed boards are collected in the diagnosis database, the statistical model will relieve the "overfitting" problem gradually. And the test merging strategy should adapt to this process. For example in $t_3 >> t_2$, if a new training case comes into the diagnosis database, e.g., $\{C_2, < 1, 1, 0, 0 >\}$ corresponding to the root-cause C_{21}. If we still allow T_1 and T_2 to be merged together, the syndrome in this new training case would be $\{C_2, < 1, 0, 0 >\}$, causing another ambiguity between C_1 and C_2. Thus, we have to cancel the previous test syndrome merging accordingly. Ideally, when the number of training cases in the diagnosis database is large enough relative to the test syndrome dimension, no test syndrome merging is needed at all.

To sum up, above example provides these intuitions:

- Test syndrome merging can relieve *overfitting* with enhanced predictability for the training cases.
- Improper test syndrome merging can bring ambiguity for the training cases to diagnose.
- Effective merging adapts to different training cases.

Therefore, it is not trivial to explore the trade-off between the predictability and the possible side-effect, rendering an essential need for an effective and efficient test syndrome merging algorithm.

3.3 Problem Formulation

According to the previous description, we formulate the problem as the following:

Given:

- the training cases;
- the structure information of board;
- the test path description

Objective:

Find N set of tests, each of which, denoted as $T_m^{'}$, is a merging of diagnostic test set $\{T_i, ...T_k\}$, such that the *Predictability* of the updated training data based on the merging is maximized with the minimum penalty of *Ambiguity* increase.

4. THE PROPOSED ALGORITHM

We can transform the problem formulation as follow: Given a graph, wherein each vertex refers to a diagnostic test, the edge between two vertices denotes a possible test merging, each edge is assigned with a weight that can reflect the gain of merging two of its ending vertices, the target is to reduce the graph size (vertex count) by merging vertices so that the gain is maximized. However, it is difficult to evaluate the predictability and the ambiguity at the same time with a single metric. Not to mention that each "merging" operation will change all the syndromes in the training cases, which change the statistical information related to all the edge weight. To decompose the above problem, We adopt a two-step approach as described in the following sections.

4.1 Heuristic to Fix the Reference Test Syndrome

In first step, we try to fix some test syndromes, denoted as *reference tests*, to be merged by the remaining test syndromes later on. The purpose, on the one hand, is to reduce the search space for test merging. On the other hand, the chosen reference tests are critical for the upcoming merging. Highly relevant fixed tests promise that the later merged tests can predict more potential faults. To guarantee the predictability in the training cases, therefore, we fix a set of tests with highly statistical relevance to the board-level components. To quantify the relevance, we employ the widely used *Mutual information*[11] as the metric. *Mutual information* measures the dependency between two events: test syndrome and component.

$$I(RC; SYN) = \sum_{c\in RC} \sum_{s\in SYN} p(c,s) \log \frac{p(c,s)}{p(c)p(s)} \qquad (1)$$

wherein the RC is the root cause on-board components and SYN is the test syndrome. The calculation is based the summation for possible value of root cause and syndrome of the joint probability multiplies by the *log* of joint probability over the marginal probabilities of each.

Next, we decide how many test items and which test items should be fixed. Greedy heuristic pursuing the highest relevance [7] is not our choice, because it may limit the predictability enhancement in the second step, leading to local optimum. For example, one component has several tests with high relevance and we fix all of them as reference tests. In the second step, however, no other tests can be merged to them, which derive far less predictability than merging them together. Therefore, a better option is to consider all the on-board components at the same time and find the reference tests using the maximal weighted matching algorithm on the basis of a bipartite graph composed of the components and the test syndromes with *mutual information* as edge weights. Noted that we exclude those test syndromes with edge as 0 in the matching results.

4.2 Heuristic for Test Syndrome Merging

In second step, we need to merge the syndromes of tests other than *reference test syndromes* (denoted as *rest test syndromes*) into the syndromes of *"reference test syndromes"*. Each merging should lead to more predictability while less ambiguity. Thus, we use the metric *ICap* defined in [2] to evaluate the gain of the merging. *ICap* defines the conditional probability of the test set to uniquely isolate a fault if the fault occurs within a component. The calculation of *ICap* for a component begins with the calculation of its ambiguous set, which is the set of functional components that are suspicious to be faulty, based on the observed syndrome λ:

$$AS(\lambda) = \bigcap_{i=0}^{f} Path(FT_i) - \bigcup_{j=0}^{p} Path(PT_j) \qquad (2)$$

where FT/PT and $Path$ refer to the failed/passed test set and the components in the testąŕs path for the given λ. We then derive the *ICap* of a component by calculating its error rate over the summation of those of all the components in the ambiguous set:

$$ICap(AS(\lambda), c_i) = ER(c_i) / \sum_{j=0}^{m} ER(c_j) \qquad (3)$$

wherein ER is the error rate of the component. The *ICap* thus can synthesize the predictability and ambiguity of a test set when all possible syndromes have been available in training cases [2]. However, if some of the syndromes may have not been observed in training cases, the efficiency of using the *ICap* is limited. Fortunately, the reference test syndromes derived in the first step ensure that any new derived syndromes by merging must have been observed in training cases. We incrementally select the test from the *rest test syndromes* that contributes the most to the *ICap* enhancement, and merge their corresponding syndromes. This greedy algorithm ends until no more *ICap* can be obtained by the merging or there is no test in *rest test syndromes*.

Above algorithm has a bottleneck: when computing the *ICap* for every tentative merging, the algorithm needs to compute the AS for many time, among which a lot of calculation is redundant. We observe that the AS union of two test set TS and TS' has the following property:

$$\begin{aligned} AS(\lambda, TS \cup TS') &= AS(\lambda(TS), TS) \cap (\cap_{i=1}^{f} P(\lambda(TS'), TS'(i)) \\ &- \cup_{j=1}^{p} P(\lambda(TS'), TS'(j))) \end{aligned}$$

where $\lambda(TS)$ is the syndrome for the tests in TS, while f and p are the number of failed tests and passed tests respectively. The above equation decompose the AS of test set union into multiple ASs of individual test set, in which a dynamic programming method can be used to reduce the redundant *ICap* calculation, by calculating the AS for every possible syndrome incrementally based on the saved immediate results of AS.

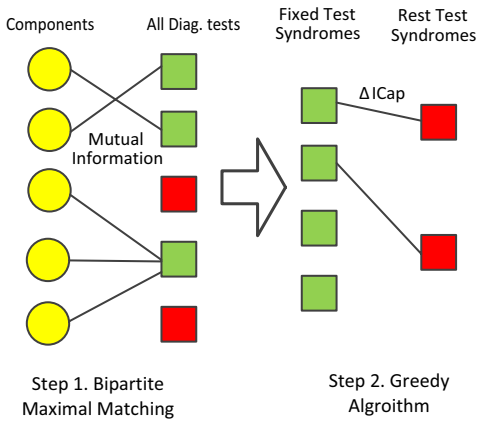


Figure 3: The proposed test merging algorithm

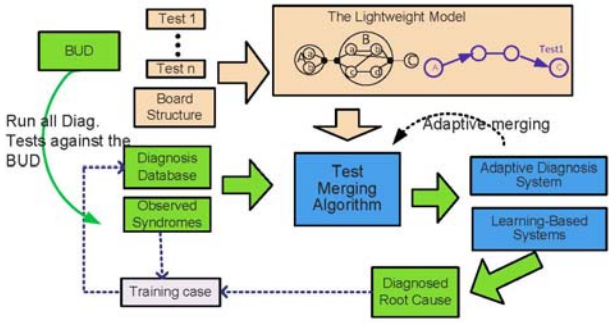


Figure 4: Overview of the proposed solution

5. REASONING-BASED DIAGNOSIS SYSTEM WITH TEST SYNDROME MERGING

In this section, we describe how to use test syndrome merging in reasoning-based diagnosis systems. We detail the integration method with two types of reasoning-based diagnosis systems as follows.

As shown in Fig. 4, the test syndrome merging algorithm takes two inputs. One is the lightweight model described in [2], while the other is the training cases from the diagnosis database. We use the proposed algorithm to merge test syndromes and apply to all the training cases in the diagnosis database. When a BUD comes, we run all the diagnostic tests and merge the observed syndromes accordingly. The updated training cases and observed syndromes of the BUD are sent to the diagnosis system (e.g., SVM) for diagnosis. After diagnosis, the successful diagnosed board is repaired and return to the manufacturer for shipment. The legacy of the diagnosis result is the pair of root-cause and test syndrome, serving as an additional training case that can enrich the diagnosis database. Therefore, the framework forms a closed loop, wherein the historical database grows in an adaptive manner. As the database becomes larger, more statistical information conveyed in the training cases. To fully use this statistical information, we can merge the test syndrome multiple times according to different scales of the database. In other words, the test syndrome merging can adapt themselves to the database, namely adaptive test syndrome merging. In this manner, we can facilitate the normal learning-based diagnosis systems(e.g. SVM, ANN, etc.) with these adaptively updated databases.

The proposed test merging solution not only can be integrated into a normal reasoning-based diagnosis system, it can be also applied in an adaptive diagnosis process (e.g., [2]). To be specific, the diangostic system iteratively selects the most

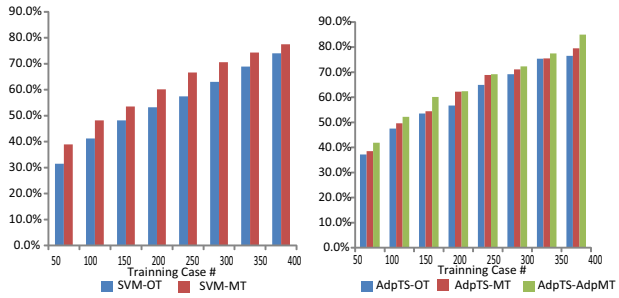


Figure 5: The SR_1 of different methods

effective tests for diagnosis and derives some suspicious components. Then, in the next iteration, it only uses those tests that are relevant to the suspicious components [1]. To integrate the proposed test syndrome merging method to such adaptive diagnosis process, for each iteration, we first apply the merged tests from last iteration to the diagnosis system to narrow down the suspicious components. Then, we fix the reference test syndromes with high relevance to these suspicious components, and continue to merge the remaining test syndrome into the fixed ones. The resulting test syndrome can be used in the next iteration. Note that we only choose the training cases that are relevant to suspicious components, rendering different training cases in different iterations. Meanwhile, the calculation of *ICap* for suspicious components also changes in each iteration. Fortunately, our test syndrome merging algorithm support this adaptive process because both of these variances are the inputs of the merging algorithm. We denote this integration as Adaptive Test Merging with doted line in Fig.4.

6. EXPERIMENTAL RESULTS

6.1 Experimental Setup

To verify the effectiveness of the proposed method, we conduct the experiment on an industrial board currently in volume production. Diagnostic team has designed a total of 499 functional diagnostic tests for it. The lightweight model is generated based on the board specification and test path information. A total of 190 repaired boards are collected from the manufacture contractor's database. Due to the limit number of historical cases, we leverage the lightweight model to generate 1000 more synthetic cases. To be specific, we randomly select a pin of a subcomponent to inject a fault into[12] and regards its on-board parent component as root-cause. After fault injection, we collect the syndrome of each test. We randomly choose a subset from above generated cases as training cases and validate the diagnostic system with the rest. To prevent the diagnosis results from being affected by different set of training cases, we have randomly selected training case for many times and report the average results.

In the experiment, we employ two baseline solutions: 1) the SVM-based diagnostic system with original test syndromes [9] (SVM-OT); 2) the SVM-based diagnosis system assisted by adaptive test selection with original test syndromes[2] (AdpTS-OT) . To evaluate the effectiveness of proposed technique, we conduct the diagnostic test syndrome merging on the training cases, and then feed the resulting training cases to above two baseline diagnosis systems, denoted as SVM-MT and AdpTS-MT respectively. At last, we integrate the proposed adaptive test syndrome merging method with AdpTS technique [2], denoted as AdpTS-AdpMT.

[1]For more details, please refer to [2]

The difference between AdpTS-MT and AdpTS-AdpMT is that the later test merging adapts to the different suspicious components in each iteration. We compare the results of above 5 methods in table 1 with varied number of training cases from 50 to 400. Successful Rate(SR) denotes the ratio of correctly diagnosed boards among the total number of diagnosed boards, and the subscript i indicates the diagnosis successful rate within i attempts. We also present Fig. 5 with only SR_1.

6.2 Results and Discussion

Table 1 shows the successful rate (SR) of three diagnosis attempts for all five solutions. The SR increases as the training cases increase. Comparing SVM-MT with SVM-OT, we observe that the test merging method improves the SR_1(Fig. 5) of a SVM-based diagnosis system by around 4% up to 10%. As the number of training cases increase, the SR benefit first increase, reach an optimal point (10%) and then drop back to 4%. The reason is that the test syndrome merging introduces more ambiguity in the ramp up stage due to the small number of training cases. When the database becomes larger, we have a better statistical property to guide the test merging. While in later stage of product cycle, the database is large enough to overcome the "overfitting" problem, rendering less benefit by test merging. When the number of diagnosis attempts increase, e.g., SR_2 and SR_3, we observe the similar trend. But, the benefit of test merging in these two scenarios are even larger than that in SR_1. That is because more attempts of diagnosis cancel out the affects of the ambiguous root-cause, and hence offset the side-effect of test merging method.

The above results provides an inspiring vision, that is, the test syndrome merging method may be more successful after integrating with any technique that can mitigate the ambitious root-cause. And that is what the adaptive test selection technique [2] is good at. To verify this idea, we integrate test syndrome merging method with adaptive test selection technique as mentioned in section 6.1, and compare it to SVM-MT. The SR gain proves that not only the adaptive test selection method can integrate with test syndrome merging method, but it also can compensate the side-effect brought by ever-increasing *ambiguity*.

One may doubt that the above SR improvement of AdpTS-MT may come from the adaptive test selection itself rather than the test merging method. For resolving the doubt and also for the fair comparison, we next compare the adaptive test selection technique with and without test syndrome merging method (see AdpTS-OT and AdpTS-MT). A similar scale of SR improvement can be witnessed, which eliminates the above doubt and proves the idea. However, we can see that the overall improvement on SR is limited, in the range of 1% to 4%. The improvement obtained by the test syndrome merging method here falls far behind what this method bring to the SVM-based diagnosis system, because the test selection process has partially mitigate the overfitting problem. Actually, both of these technique has a common limitation, wherein the statistical information of the historical data is omitted during test syndrome selection and merging. In another word, they are "not adaptive enough".

As a result, we apply the adaptive test selection with adaptive test syndrome merging, i.e., AdpTS-AdpMT. Compared to AdpTS-MT, this method can further outperforms up to 5% improvement on SR.

It worth noting that AdpTS-AdpMT with 200 training cases can compete with SVM-OT with 300 training cases. It therefore highlights the ability of the proposed test syndrome merging method to overcome the overfitting problem due to the lack of training cases in product ramp up stage.

To compare our method with statistical feature selection method, we try an advanced statistical feature selection method, minimum-redundancy-maximum-relevance(mRMR), mentioned

Train case #	SVM-OT			AdpTS-OT			SVM-MT			AdpTS-MT			AdpTS-AdpMT		
	SR_1	SR_2	SR_3	SR_1	SR_2	SR_3	SR_1	SR_2	SR_3	SR_1	SR_2	SR_3	SR_1	SR_2	SR_3
50	*31.5%*	35.2%	38.4%	37.2%	40.1%	44.5%	38.9%	41.2%	50.2%	38.5%	42.2%	47.7%	41.9%	43.3%	50.2%
100	*41.2%*	46.3%	48.5%	47.5%	50.6%	53.6%	48.2%	52.0%	55.7%	49.6%	53.3%	54.4%	52.2%	55.5%	61.1%
150	*48.2%*	53.5%	57.4%	53.5%	56.0%	60.3%	53.5%	55.5%	64.2%	54.6%	55.5%	60.5%	60.1%	64.2%	66.0%
200	*53.2%*	56.4%	59.3%	59.3%	61.1%	64.9%	60.1%	61.1%	66.7%	62.2%	63.7%	66.2%	62.8%	66.7%	70.0%
250	*57.4%*	61.7%	65.0%	64.9%	65.4%	69.9%	66.7%	68.8%	72.2%	68.9%	71.2%	72.9%	69.2%	71.1%	75.5%
300	*63.0%*	66.1%	71.2%	69.2%	71.7%	77.7%	70.6%	72.3%	77.0%	71.1%	71.7%	76.7%	72.3%	75.0%	78.2%
350	*68.9%*	72.3%	75.6%	75.4%	76.6%	81.6%	74.3%	77.7%	83.3%	75.5%	78.3%	81.2%	77.5%	81.5%	85.5%
400	*74.3%*	78.2%	83.3%	76.5%	82.2%	88.3%	77.5%	81.0%	90.0%	79.5%	81.2%	88.3%	85.5%	90.0%	92.0%

Table 1: Diagnosis accuracy of different diagnostic methods under varying training Cases

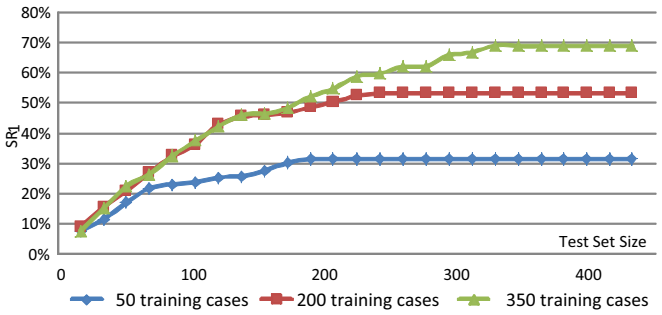


Figure 6: mRMR feature selection method

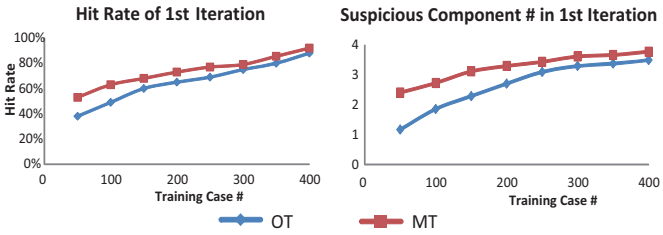


Figure 7: Hit rate and the size of suspicious components in the 1^{st} iteration

in [7] as a speed-up method, which targets a different objective from us. We plot the curve between the number of selected tests and the SR in Figure 6 with 50, 200 and 350 training cases. The tests are incrementally selected based on the ranking of the mRMR metric in descending order defined in [7]. The curve shows a stable improvement on SR with the increase of the number of tests, which indicates that selecting any subset of the whole test set with the method will lead to diagnostic accuracy decrease, since there is information loss. However, as stated, we can improve SR with our test merging method, which demonstrates that the test merging is more preferable for feature reduction in this scenario.

An interesting observation in the table is that there is no obvious winner beween SVM-MT and AdpTS-OT. In order to look into the reason, as can be seen in Fig 7, we plot the "hit rate", the ratio of the root cause in the suspicious components (Fig 7(a)) and the the average number of the suspicious components(Fig 7(a)) in the first iteration. Noted that all methods, which we mentioned in experimental results, with MT has the same "hit rate" in the 1st iteration, and so does the both methods of OT. Obviously, the adaptive test selection with merged test set always has higher SRs, proving that the test syndrome merging preserve more information conveyed in diagnostic test syndromes. However, the improvement of hit rate keeps decreasing from more than 10% to 3% as training cases increases.The results indicate that the merging improves the probability that the root-cause component becomes in the suspicious components, but originally they are not. The improvement becomes less when more training cases available, since the large number of training cases themselves have high probability to detect the suspicious component out and the effect of test merging becomes gradually less obvious. The same trend can also be noticed for the average number of the suspicious component in first iteration, as can be seen in Fig 7(b). The MT method has an increase average number of suspicious components, which is from around 1.2 to around 0.2 with the increase of training cases. Also more importantly, we note that, compared to OT methods, the MT methods always has more suspicious components, which is brought by the *"ambiguity"* increase.

7. CONCLUSION

In this paper, we propose a novel test merging methodology to reduce the feature size in reasoning-based diagnostic systems, by leveraging the domain knowledge of the board under test and the corresponding diagnostic test set, such that the available training cases can be effectively used to improve diagnosis accuracy. Experimental results highlight the effectiveness of the proposed method.

8. REFERENCES

[1] K. P. Parker. Defect coverage of boundary-scan tests: what does it mean when a boundary-scan test passes? In *Proc. IEEE International Test Conference (ITC)*, pp. 181–189, 2003.

[2] Z. Sun, L. Jiang, Q. Xu, Z. Zhang, Z. Wang, and X. Gu. AgentDiag: An agent-assisted diagnostic framework for board-level functional failures. In *Proc. IEEE International Test Conference (ITC)*, pp. 1–8, 2013.

[3] T. Satyanarayana, G. Subramanyam, and K.V. Rama Rao. Implementing an expert system for fault diagnosis of electronic equipment. *Engineering Applications of Artificial Intelligence*, 8(3):355–364, 1995.

[4] B. Chess and T. Larrabee. Creating small fault dictionaries. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 18(3):346–356, 1999.

[5] Z. Zhang, K. Chakrabarty, Z. Wang, Z. Wang, and X. Gu. Smart diagnosis: Efficient board-level diagnosis and repair using artificial neural networks. In *Proc. IEEE International Test Conference (ITC)*, pp. 1–9, 2011.

[6] D. M. Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[7] F. Ye, Z. Zhang, K. Chakrabarty, and X. Gu. Information-theoretic syndrome and root-cause analysis for guiding board-level fault diagnosis. In *Proc. IEEE European Test Symposium (ETS)*, pp. 1–6, 2013.

[8] G. Brown, A. Pocock, M.J. Zhao, and M. Luján. Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. *The Journal of Machine Learning Research*, 13:27–66, 2012.

[9] Z. Zhang, X. Gu, Y. Xie, Z. Wang, Z. Wang, and K. Chakrabarty. Diagnostic system based on support-vector machines for board-level functional diagnosis. In *Proc. IEEE European Test Symposium*, 2012.

[10] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[11] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[12] Z. Zhang, Z. Wang, X. Gu, and K. Chakrabarty. Physical defect modeling for fault insertion in system reliability test. In *Proc. IEEE International Test Conference*, pp. 1–10, 2009.