IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 20, 2020, accepted August 7, 2020, date of publication August 12, 2020, date of current version August 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3016039

Pattern Reorder for Test Cost Reduction Through Improved SVMRANK Algorithm

TAI SONG[1], HUAGUO LIANG[1], TIANMING NI[2], (Member, IEEE), ZHENGFENG HUANG[1], YINGCHUN LU[1], JINLEI WAN[1], AND AIBIN YAN[3], (Member, IEEE)

[1]School of Electronic Science & Applied Physics, Hefei University of Technology, Hefei 230009, China
[2]College of Electrical Engineering, Anhui Polytechnic University, Wuhu 241000, China
[3]School of Computer Science and Technology, Anhui University, Hefei 230601, China

Corresponding author: Huaguo Liang (huagulg@hfut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61674048, Grant 61874156, and Grant 61904001; in part by the Anhui Provincial Natural Science Foundation under Grant 1908085QF272; and in part by the University Synergy Innovation Program of Anhui Province under Grant GXXT-2019-030.

ABSTRACT With the growing complexity of integrated circuits (IC), more and more test patterns are added to test set to test more defects, making the number of test pattern and individual test pattern length continues to increase as the size of IC gets larger, boosting test time and consequently test cost. To solve this problem, this paper proposes a kind of valid pattern identification method. The method uses machine learning to reorder the test pattern which can select the most effective patterns, to determine the optimal training set and test set first. Then, by means of the weighted SVMRANK algorithm to find the optimal pattern sequence. Experiment results demonstrate that the method only sacrifices 2% prediction accuracy in exchange for 3.89 times the time saving. The method aims at maximizing the accuracy of test, and minimizing the number of patterns. The proposed idea significantly improves the test time and test efficiency compared to conventional test flows. This is an innovative test cost reduction method with the growing complexity of IC.

INDEX TERMS Adaptive test, machine learning, test reorder.

I. INTRODUCTION

Very Large Scale Integration test is an important and first step for identifying defects within a failing chips. To address this problem, a large number of test patterns (referred to as ''test items'' in this paper) are needed to ensure that the Defect Part Per Million (DPPM) is less than one [1]. In traditional testing, all test items are applied to the test, and the test set (test content or patterns) is applied to each circuit until it passes the whole suite of tests or fails one of the tests. A chip is considered as pass only if it passes all the test items. Based on the test results, each chip is classified as being either pass or fail. The failed chips are either discarded or forwarded for diagnosis and failure analysis, while the pass chips are sent to the next stage of testing.

The standard test approach is straightforward to put in place and results in high test quality. However, it incurs very high test cost since it requires sophisticated Automatic Test Equipment (ATE) and long test time [2]. According to International Technology Roadmap for Semiconductors (ITRS) data, test cost is now a major part of the manufacturing cost [3], which has emerged as a major showstopper for further advancement. Therefore, there is a pressing need to find a new test method to reduce the test cost, and the corresponding scientific issues are summarized as follows:

1. With traditional test approach, all parts are tested the same, regardless of the individual performance of each part. The approach is somewhat satisfactory when variation is small, but when large variation exists, traditional methods will be insufficient. For example, either an excessive number of good dies are failed due to overly tight limits to ensure defective dies are detected, or there are an excessive number of test escapes (i.e., faulty circuits are undetected) due to overly loose limits to reduce yield loss (i.e. non-faulty circuits that are inadvertently discarded) [4].

2. The rapid growth of the silicon process over the last few decades has significantly improved semiconductor integration levels. The transistors today are smaller, faster and cheaper than ever before. However, this aggressive downscaling of dimensions in the forthcoming CMOS technology generations poses critical testability issues [5], especially the test cost keeps increasing as technology nodes advance.

The associate editor coordinating the review of this manuscript and approving it for publication was Stavros Souravlas.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

3. If the chips to be tested come from the same fabs, the same batch, and the same flow, then the defects of this batch may be the same, in this case, there is no need to use all test patterns, only a small number of corresponding test items can detect these defects [5]. However, traditionally, the test limits, test content and test flows are statically set and only occasionally updated based on expert experience, and the time scale is typically every few months, so that there is pressing need to find a new test method to remove ineffective patterns depending on defect characteristics.

4. Some irrelevant or redundant test items exist in the increased test pattern, and the effectiveness of the test pattern is reduced [6]. Production test suites include a large number of redundant test patterns due to the inclusion of multiple test types with overlapping defect detection and the use of simple fault models for test generation [7]. In addition to ever increasing circuit sizes, the need for new test types constitutes one of the main drivers of test pattern increase [8]. New test types such as N-detect [9] and gate-exhaustive tests [10] further exacerbate the issue by systematically adding redundancy into the test set in order to increase the probability of detection of actual silicon defects. Therefore, reducing redundancy of test set becomes increasingly critical because of the multiplication of data.

5. Enormous defect detection overlap problem between test types has not yet received commensurate attention. IBM has historically found that 70%-90% of the logic test patterns can be removed for Application Specific Integrated Circuit (ASIC) without any impact to test escapes shipped to customers [11]. As previously reported by industry and academic researchers, a significant number of redundant patterns exist in the test sets, resulting in a substantial increase in test time yet at no concomitant defect coverage benefit.

6. From another point of view, advanced semiconductor processes allow design specifications to tolerate parameter variations in the manufacturing process stage, to achieve acceptable yields, sometimes this will lead to potential manufacturing defects, typically expressed in terms of test escapes. Such undetected defect may lead to chip failures and reduce reliability, leading to early life failures. Therefore, there is a pressing need to find a new test method to improve reliability.

From the abovementioned scientific issues, with the growing complexity of IC, testing needs more and more test patterns, the number of test patterns and individual test pattern length continue to increase as the size of IC gets larger, and thus, boosting test time and consequently test cost [12]. The objective of test is to cost effectively screen parts for high quality and reliability, while at the same time limiting unnecessary yield loss. It is extremely difficult to do this using traditional methods. To improve the performance of test, some forms of adaptation is essential.

In recent years, there has been an intense effort to develop alternative test approaches that can replace effectively the standard test approach. Therefore, an adaptive testing strategy is now being advocated [2]. The concept of adaptive testing is to divide the chip into multiple groups, randomly selects a small set of chips as sample chips for each group, and then applies a complete set of testing to these sample chips. Based on the test outcomes of the sample chips, adaptive testing selects only the most effective test items for each group. In this way, significant test time savings and test quality improvements expected can be achieved.

In adaptive testing, each wafer or die can be tested using a unique test process. Adjusting the test process may involve adjusting: (a) The test content, i.e. given a test suite certain ineffective patterns are dropped so as to save test time; and (b) The test order, tests with higher fail rate are moved forward and applied first so as to save test time; and (c) The test limits adjustment so as to improve outlier detection and quality control [2]. This work uses the method to reorder the test patterns to achieve the purpose of saving test costs.

Considerable amount of effort has been expended on test pattern reordering schemes that aim at improving the test time, and many works based on adaptive test methods have been published in the field of testing. In [13], [14], authors use the fault coverage metric to approximate pattern effectiveness in pattern reordering, and investigate the impact of reordering four different types of tests (functional, IDDQ, stuck-at and delay). However, the method may not be as effective for catching defects. In order to address this problem, a somewhat adaptive and finer grained test reordering approach in [15] analyzes a sample set of failed devices, identifies which devices fail which tests, and computes an optimal order of tests via dynamic programming or heuristics. However, it may not be effective in detecting effective chips quickly. Although appreciable benefits have already been observed in established work in [8], no algorithmic approach has been proposed to efficiently utilize the correlation information in adaptive test development, hindering the full exploitation of the potential of adaptive techniques. Finally, the adaptive test technique proposed in [7] continuously reorders tests based on the test fail rates, which is the actual effectiveness measure for tests. However, the method didn't consider the trade-off between test time and test accuracy.

As we can see, most of previous works are based on minimizing test cost [16], [17], they don't consider the test quality, at the same time, increased hardware overhead leads to increased manufacturing costs, and lacking research on screen out the most effective test patterns, which can provide test cost reduction without increasing the defect level. To bridge this gap and formally show that, in a quest to reduce test cost and increase test quality, the work presented in this paper adopts improved SVMRANK as a pattern selection algorithm which can select the most effective patterns while guaranteeing the test quality. This paper presents an innovative pattern reorder approach to select the optimal test patterns as is shown in Fig.1.

The main contributions of this paper are shown as follows:

1) We use machine learning algorithm to rank and reorder the test patterns and consequently reduce test time.

2) We proposed an improved SVMRANK method with weight function to improve defect prediction accuracy.

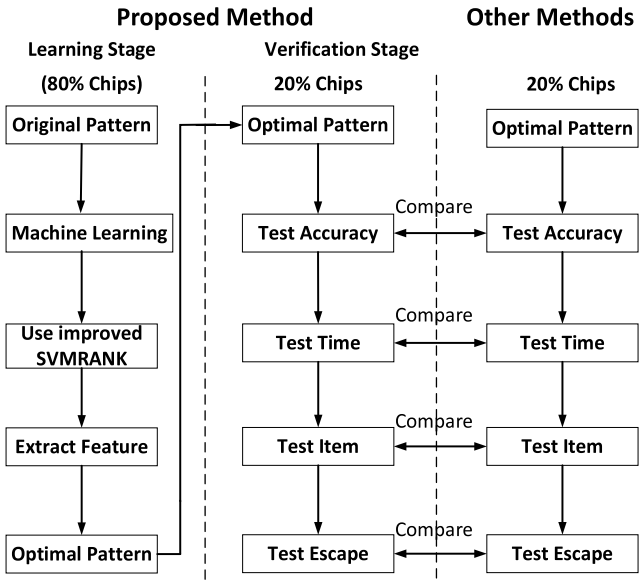

FIGURE 1. Test reorder adaptive test flow.

The rest of the paper is structured as follows. Section II presents the background and test strategy. In Section III, the proposed reorder algorithm is presented in detail. Section IV shows experimental environment and verification results, and the conclusions are drawn in Section V.

II. BACKGROUND

A. ADAPTIVE TEST DEFINITION

As for traditional test approach, the test content, test order and test limits are fixed, the test set is applied to each circuit until it passes the whole suite of tests or fails one of the tests, and it follows a fixed procedure. However, if there is only a small part of test patterns are valid, and still use conventional 100% test patterns, that obviously wastes of test time. Therefore, there is a pressing need to find a new test method to reduce test cost.

Adaptive testing is a promising approach that practically ensures cost reduction (eliminate redundancy and use fewer test patterns) and test quality (fewer test escape) for testing strategy. In adaptive test, the test content, test order and pass/fail limits are not fixed as in conventional test, but depend on other test results of the currently or historically tested data (training data), and data analysis that can be used to adjust test limits and content during production testing on-the-fly. Adaptive testing [16], [17] is now being advocated, by dropping unnecessary test items to reduce the test costs, or by heightening test limit criteria to improve the test quality. Unlike conventional test, all test items do not require thorough testing, it considers whether these test items are effective based on the test outcomes of the sample dies, and adaptive testing selects only the most effective test items. In this way, significant test time savings or test quality improvements can be achieved compared with traditional test flows.

B. TEST REORDER UNDER STOP-AT-FIRST-FAIL MECHANISM

In ATE test, where stop-at-first-fail is employed. The idea is to apply more effective patterns/tests earlier to make defective chips quickly fail. Fig. 2. shows the test reorder approach. Item 2 composes of five test contents, the first two test content pass and marked as P, and the third test fails marked as F. Then, the fourth and fifth tests are no longer executed, it means that test item 2 stops, and this is stop-at-first-fail, which saves two fifths of test time. Apparently, such schemes help reduce test time for only the dies that fail the test, as all tests end up being applied on passing dies. In addition, stop-at-first-fail is not only used to save test time, but to protect the ATE hardware.

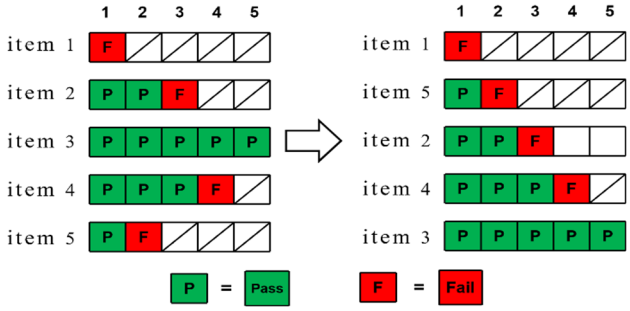

FIGURE 2. Test reorder flow under stop-at-first-fail mechanism.

In Fig. 2 the input data of test set consists of 5 tests (item 1, item2, ..., item5) that are called here the original set (the left part in Fig. 2). Test item 1 to 5 are executed sequentially as traditional test method. In order to achieve the biggest time savings, adaptive test adopts reordering test item method (the right part in Fig. 2), rank forward the item which are most likely to fail, and the most effective test item can be executed first.

C. ADAPTIVE TEST: SEARCHING THE OPTIMAL SUBSET DATA

Multiple subsets of test patterns may have the redundancy problems. Feature (pattern) selection is one of the effective ways to solve the problems. The significance of feature selection is to remove some irrelevant subsets or duplicate subsets, and reduce the number of subsets while ensuring the validity of the subsets, reducing test patterns complexity and test time. Hence, the objective of feature selection is to improve the test performance by avoiding the interference of redundant and irrelevant features, to provide faster and more effective test pattens and to help the data processing by enabling a more parsimonious representation of the test sets. Adaptive test can shorten the test time by reducing or reordering the test contents as is shown in Fig. 3.

Although the test time for a good die is fixed no matter what size is used for applying the tests, the test time for a faulty die may vary greatly for distinct test size. The approximation of the ideal test size represents essentially a search for the

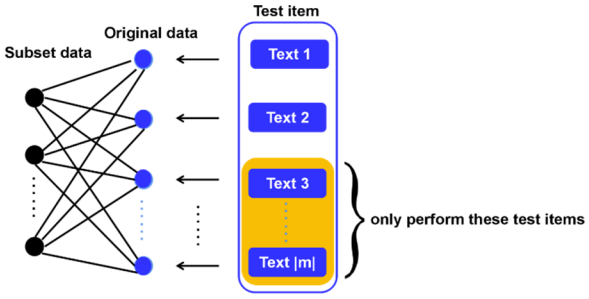

FIGURE 3. Searching for optimal subset.

maximization of the early defect detection probability. Attaining such a goal requires mathematical models that predict the capability of a particularly test in detecting the defects that have not yet been covered by the tests before it, in turn again necessitating an accurate correlation analysis between tests.

III. PROPOSED IMPROVED SVMRANK METHOD

In test patterns ranking, the basic mechanism of SVMRANK is to convert the patterns ranking problem to binary classification problem of each candidate pair. Then, the support vector machine (SVM) will be utilized for the binary classification of multiple patterns, so as to achieve ranking. All candidate items are ranked according to their importance features.

A. CONVERT THE RANKING PROBLEM TO A CLASSIFICATION PROBLEM

First, we define a feature vector: x. The ranking function is $f(x)$, and then determine which test item is ranked first and which item is ranked next based on the size of $f(x)$. That is, if $f(x_i) > f(x_j)$, then x_i should be ranked before x_j, and vice versa. It can be expressed by the following formula:

$$x_i > x_j \Leftrightarrow f(x_i) > f(x_j) \tag{1}$$

In theory, $f(x)$ can be any function. For simplicity, this work assume that it is a linear function:

$$f(x) = \langle w, x \rangle \tag{2}$$

If this ranking function $f(x)$ is a linear function, then it can be converted a ranking problem to a binary classification problem. The reasons are as follows:

First, for any two feature vectors x_i and x_j, under the premise that $f(x_i)$ is a linear function, the following relationship exists:

$$f(x_i) > f(x_j) \Leftrightarrow \langle w, x_i - x_j \rangle > 0 \tag{3}$$

Then, the binary classification problem can be considered as the difference vector of x_i and x_j. We can assign a label to the difference:

$$y = \begin{cases} +1, & if\ x_i - x_j > 0 \\ -1, & if\ x_i - x_j < 0 \end{cases}$$
$$\langle w, x_i - x_j \rangle > 0 \Leftrightarrow \mathrm{y} = \mathrm{y} + 1 \tag{4}$$

B. USING SVM MODEL TO SOLVE THE RANKING PROBLEM

After converting the ranking problem into a classification problem, it can use the commonly used classification model to learn. Here this work chooses Linear SVM. Similarly, it can be extended it to nonlinear SVM through the kernel function method.

As shown in the Fig. 4 below, it is an example of a ranking problem. There are two sets of queries and their corresponding recalled documents. Where the relevance level of documents is divided into three ranks. The weight vector w corresponds to the ranking function $f(x) = \langle w, x \rangle$, and the query pair can be scored and sorted.

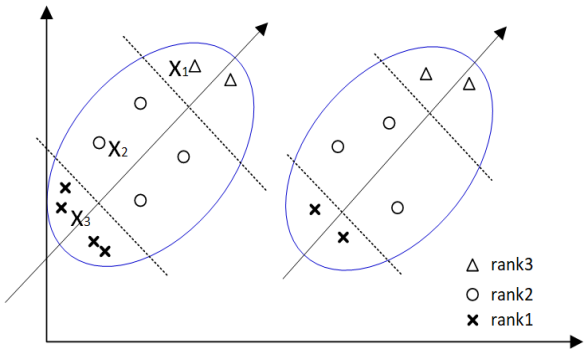

FIGURE 4. Ranking problem.

The following Fig. 5 shows how to convert the ranking problem into a classification problem. The feature vectors of pattern with different relevance levels in the same group (under the same query) can be combined to form new feature vectors: $x_1 - x_2$, $x_1 - x_3$, $x_2 - x_3$. Similarly, the label will be re-assigned. For example, the labels of the feature vectors such as $x_1 - x_2$, $x_1 - x_3$, and $x_2 - x_3$ are assigned as the positive labels in the classification problem. Further, in order to form a standard classification problem, we also need negative samples. Here this work uses the reverse vectors of the new positive feature vectors as corresponding negative samples: $x_2 - x_1$, $x_3 - x_1$, $x_3 - x_1$. In addition, it should be noted that when we combine to form a new feature vector, it cannot use two feature vectors at the same similarity level in the original ranking problem, nor can we use two feature vectors under different queries.

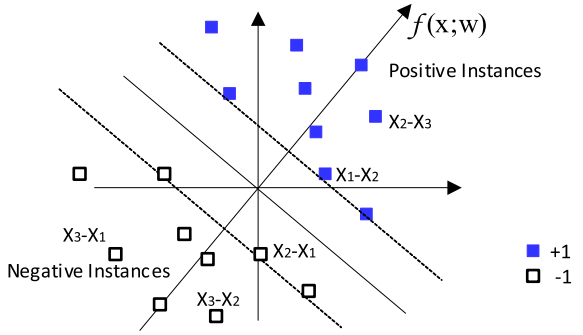

FIGURE 5. Classification problem.

C. SOLVING PROCESS OF SVM MODEL

After transforming into a classification problem, it can be used the general method of SVM to solve classification, and mark the classes of all relevant candidate pairs $\left(x_i^{(1)}, x_i^{(2)}\right)$, where $x_i^{(1)}$ is the first candidate of the i-th pair, and $x_i^{(2)}$ is the second one. To achieve the binary classification target for the built dataset, the SVM is established as follows:

$$\begin{aligned} &min_{w,\varepsilon}\frac{1}{2}\|w\|^2 + C\Sigma_{i=1}^{N}\varepsilon_i \\ &s.t.\ y_i\left\langle w, x_i^{(1)}, x_i^{(2)}\right\rangle \geq 1-\varepsilon_i \\ &\quad \varepsilon_i \geq 0 \\ &\quad i = 1, \ldots, N \end{aligned} \tag{5}$$

wherein, C represents the penalty factor, ε_i denotes slack variable, and $y_i \in \{0,1\}$ signifies the class of i-th pair.

By introducing constraints into the relaxation variables of the original optimization problem, it can be further transformed into an unconstrained optimization problem:

$$\min_{w} \sum\nolimits_{i=1}^{N} [1 - y_i\left\langle w, x_i^{(1)}, x_i^{(2)}\right\rangle]_+ + \lambda\|w\|^2 \tag{6}$$

The first term of the sum represents the hinge loss, and the second term represents the regular term.

D. IMPROVED SVMRANK METHOD

The above part is the introduction of the basic SVMRANK classification algorithm. The following is to find the optimal solution by modifying the loss function. In SVMRANK, the optimization goal is better consistent with the commonly used evaluation index of information retrieval problem, specifically, it needs to give different weights to distinguish, in addition, it uses cost sensitive classification instead of 0-1 classification, the usual hinge loss is modified to give different loss weights.

1) For a pair with a higher similarity level, a larger loss weight is assigned.

2) For a smaller number of queries, a larger loss weight is given.

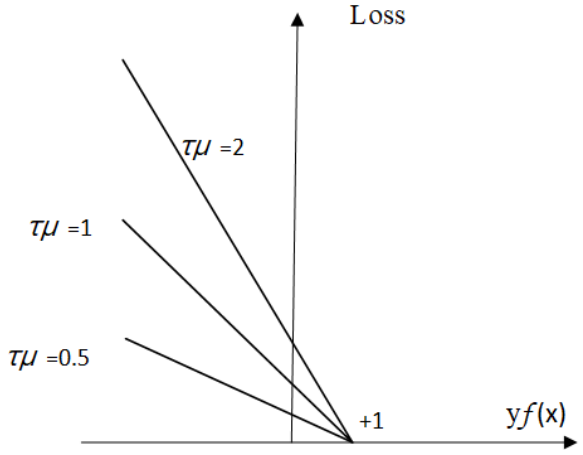

FIGURE 6. Loss weight function.

The optimization problem of SVMRANK can be expressed as follows:

$$\min_{w} \sum_{i=1}^{N} \mathfrak{C}_{k(i)}\mu_{q(i)}[1 - y_i\left\langle w, x_i^{(1)}, x_i^{(2)}\right\rangle]_+ + \lambda\|w\|^2 \tag{7}$$

where $\mathfrak{C}_{k(i)}$ represents the loss weight value of the instance belonging to the k-th grade pair. And average all the reduction values to get this loss weight. It is conceivable that the greater the loss weight value, the greater the influence of the pair on the overall evaluation index, so the importance of training is correspondingly greater. This is to make the training result focus on the ranking position, and vice versa.

Because the relative importance of SVMRANK training will be low, the improved method can increase the importance of the pattern pair under the query by increasing the weight parameter, so that the model training can attach considerable importance to the doc pair under different queries. Therefore, the optimization problem of SVMRANK is as follows:

$$\begin{aligned} &min_{w,\varepsilon}\frac{1}{2}\|w\|^2 + C_i\Sigma_{i=1}^{N}\varepsilon_i \\ &s.t.\ y_i\left\langle w, x_i^{(1)}, x_i^{(2)}\right\rangle \geq 1-\varepsilon_i \\ &\quad C_i = \frac{\mathfrak{C}_{k(i)}\mu_{q(i)}}{2\lambda} \\ &\quad \varepsilon_i \geq 0 \\ &\quad i = 1, \ldots, N \end{aligned} \tag{8}$$

The correlations among multiple importance evaluation criteria are generally non-linear. It is difficult for the linear classifier to capture the non-linear relationship among various features. In the application of SVM to address the non-linear problems, kernel functions can be used to map the feature space from the low-dimensional space to the high-dimensional space. Subsequently, the non-linear problems may be solved by using linear classifier in the high-dimensional space.

IV. EXPERIMENTAL RESULTS

A. FAULT SIMULATION ENVIRONMENT SETUP

The simulation experiment is conducted according to reference [22], [23] as is shown in Table 1. Due to the absence of actual defect detection data, the simulation test patterns reduction may vary based on real defect behavior and test types used in the production.

To better mimic reality, a pool of defective circuits is generated by randomly injecting cell defects, open defects, and bridge defects, one at a time, into a layout implementation of each benchmark [19]. The cell defects injected include internal line opens, bridges between internal lines, bridges between inputs of a cell, feedback bridges between the inputs and output of a cell, transistor stuck-opens, transistor stuck-close defects [20]. For each layout with an injected defect, the corresponding circuit-level [21] netlist is extracted and circuit-level simulation is performed using a test set that achieves 100% stuck-at fault efficiency, and the circuit

TABLE 1. Preliminary number of defective circuits.

Benchmark	Gates	Patterns	Circuits
s5378	1004	258	5000
s9234	2017	379	5000
s13207	2573	480	5000
s15850	3448	437	5000
s38417	8709	919	5000
s38584	11448	653	5000

responses are digitized and collected. In order to simulate industrial design, by means of iterative method consisting of 48,500 flip flops and 1.1 million gates are used in the experiments. 4,000 stuck-at scan vectors are generated by a commercial ATPG tool.

B. DETERMINE RATIO

In order to select the best ratio of training set and test set, and find the optimal value of ranking result, therefore, the training data are selected randomly but different from each other, and the test data are collected from the rest of the sampled data. The ratio of the size of training data to that of the test data are 1:9, 2:8, 3:7, 4:6, 5:5, 6:4, 7:3, 8:2, and 9:1. The resampling is repeated 10 times. As is shown in Fig 7. The result might expect that the larger the size of training data, the smaller the value of MSE for test data.

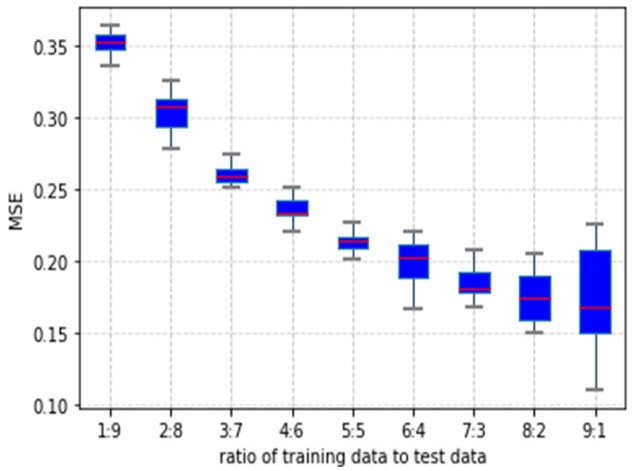

FIGURE 7. Comparison MSE of various ratios.

Fig 7. Shows that the value of MSE is preferably at 0.17 and ratio at 8:2. it is obvious that the ratio of 8:2 is more robust than 9:1. Therefore, 80% of test patterns as training data and 20% of test patterns as test data were chosen in this work.

To evaluate the efficiency and accuracy of the proposed method, the experiment is performed on the test pattern generated by simulation results. Test simulation program is implemented using Python. Machine learning experiment platform adopted python 3.7 and scikit-learn library, test_size = 0.2, and random_state = 0.

C. PREDICTIVE ACCURACY COMPARISON

In order to compare the accuracy of the proposed method, four different test flows were used. The ratio of the size of training data to that of the test data are 8:2. The resampling is repeated 10 times and get the average value of accuracy. Fig. 8. shows the predictive accuracy. The method uses traditional method, the proposed method (with weight), SVMRANK method (no weight), and the test-set reordering method [7] are compared in Fig. 8 respectively. The predictive accuracy of the proposed method (which is 89%) is 2% lower than the traditional method (which is 91%), but higher than SVMRANK method (which is 83%) and test-set reordering method [7] (which is 82%). From the results, it can be seen that the accuracy of the proposed method is better than SVMRANK and the method [7].

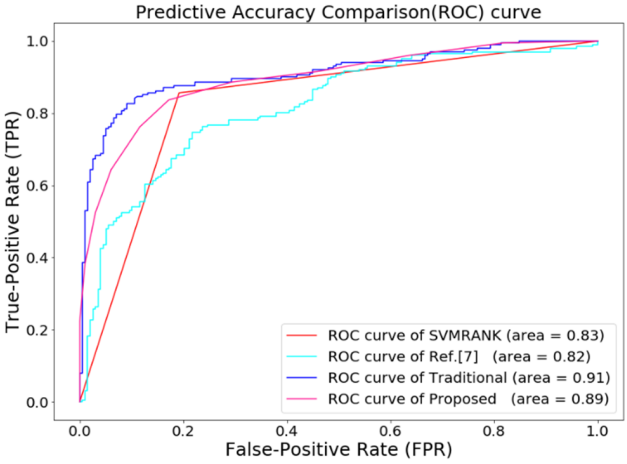

FIGURE 8. Predictive accuracy receiver operating characteristic (ROC) curve.

D. TEST TIME COMPARISON

In order to compare the test time of the proposed method, 200,000 faults that are randomly selected from the sample set of stuck-at faults are injected to the design one at a time [23]. The experiments start with a random test order. The trailing average of test time for the last 10,000 faults at each point during the testing is reported in Fig. 9.

Since each pattern has the identical length in our experiments, the test time reported is proportional to the number of patterns applied. Fig. 9. shows that the proposed method (with weight) has the smallest testing time.

In order to compare the test time of proposed method and the traditional method and choose the optimal proportion of the test patterns, Table 2 shows a close examination of the results shows that detection time scoring (150,000 faults and 200,000 faults) delivers better results than detection count scoring (50,000 faults and 10,000 faults). The improved SVMRANK test method with detection time scoring (250.47 ÷ 64.32 = 3.89X) delivers the highest reduction

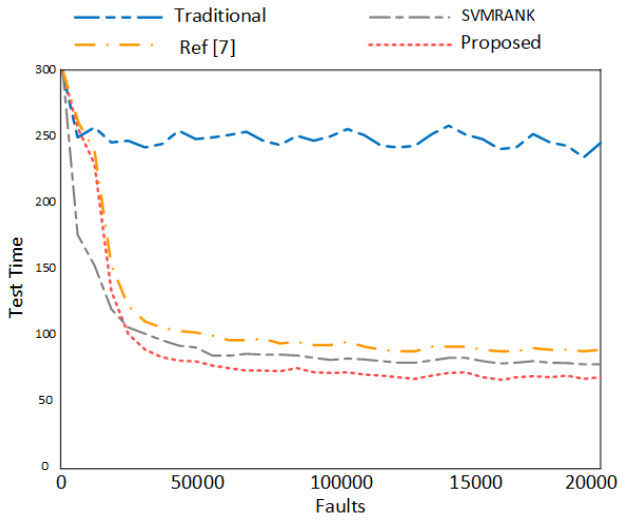

FIGURE 9. Average test time comparison.

in test time, which shows that the more test patterns, the greater the test time savings.

E. TEST ITEMS COMPARISON

In our application, the test items have similar test time. In order to compare the selected test items to each other, the experiment is repeated 5 times to get the value of test items, and the overall comparison result is shown in Fig. 10, which shows that the proposed method always has the smallest test items.

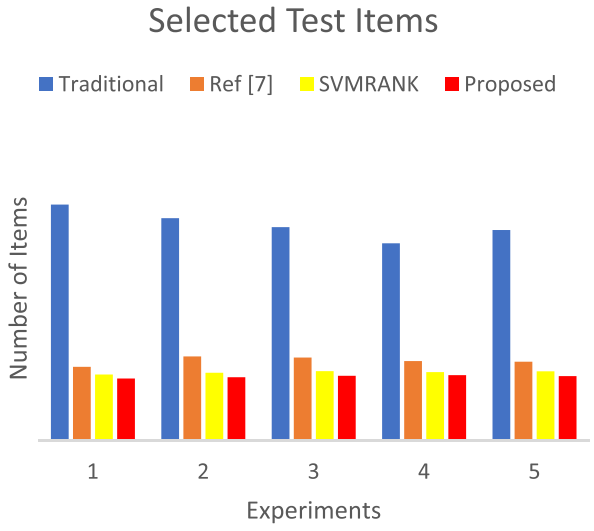

FIGURE 10. Select test item comparison.

F. TEST ESCAPE COMPARISON

In order to compare the test escape of the proposed method, 10% to 100% of the test items were used. The resampling is repeated 10 times to get the average value of test escape.

Fig. 11 shows the proposed method has the smallest number of test escapes compared to the other three methods.

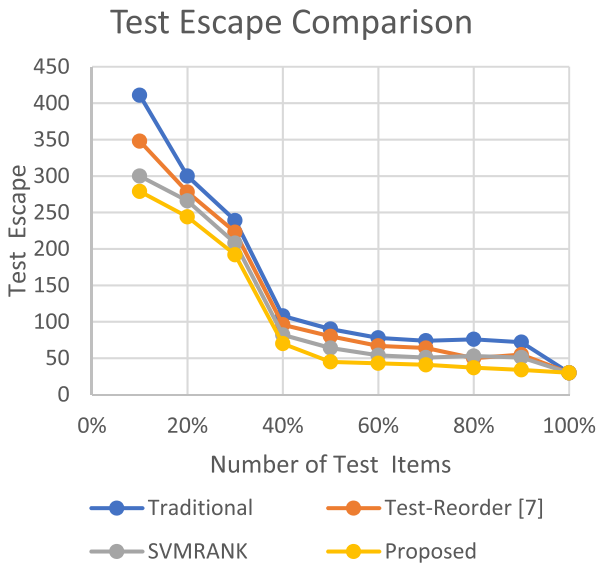

FIGURE 11. Test escape comparison.

In order to compare overall result and prove that our method is better than the other methods, the proposed method is compared with the conventional test scheme and the test reorder method proposed in [7], in terms of test cost (test time) and test quality (predictive accuracy). The compare results for the improved SVMRANK techniques (with weight) are listed in Table 2 when proportion of test pattern is 20%.

TABLE 2. Test time improvement over the traditional test.

Test Time (sec.)	50000 faults (74.23)	100000 faults (74.57)	150000 faults (69.67)	200000 faults (64.32)
traditional (250.47)	3.37X	3.35X	3.59X	3.89X

The last column of Table 3 shows the comparison between the proposed technique and the approach in [7]. Distinct benefits of the proposed technique over the previous adaptive approach can be observed, both in terms of test time and quality. From Table 3 it can be calculated that the proposed method saves test time to 10.6% and the test quality is improved to 8.5% compare with the approach in [7].

In order to show the superiority of the proposed method, this work compared the pattern reordering method in the last three years. In terms of accuracy, test time, select test items and the number of test escape. The compare results are shown in Table 4.

From Table 4 we can see the reorder-based [2] and [23] shown that the test time equivalent to test cost are less than the proposed method, but there are more test escapes than proposed method, they only consider the reduction of test cost and ignore the increase of test escape. The reorder-based Ref [7] and Ref [12] have the same test escape as proposed method, experimental results show that they only focus on

TABLE 3. Comparison of other test flow.

Method / Performance	Traditional	Method [7]	SVMRANK	Proposed	Improve Over [7]
Predictive accuracy	91%	82%	83%	89%	8.5%
Test time (sec.)	250.47	72	69	64.32	10.6%
Select test Items	1730	648	569	530	18.2%
Test escape	76	50	53	37	26%

TABLE 4. Comparison of other reordering methods.

Performance / Methods	Predictive accuracy (%)	Test time (sec.)	Select test Items	Number of Test escape
2018 reorder [2]	87	59.4	612	52
2017 reorder [7]	82	72	548	40
2019 reorder [12]	86.4	68.13	531	38
2018 reorder [17]	88	65	534	38
2020 reorder [23]	87	59.24	552	80
Proposed	**89**	**64.32**	**530**	**37**

test quality, but ignore the test cost. The most similar result is to based Ref [17] method, however, it is the reduction of test time and test escape by increasing a large amount of hardware circuit and area overhead, and the proposed method is completely software-based and does not require any additional hardware overhead. Therefore, it is the best compromise between test escape and test cost of proposed method.

V. CONCLUSION

Our work is an innovative test cost reduction method, which selects the effective test patterns with improved SVMRANK algorithm. The advantage is that it will make test patterns with higher fail rate are moved forward and applied first so as to save test time, and comparison show that the proposed idea significantly improves the test time and test efficiency compared to conventional test flows, which can provide test cost reduction without increasing the defect level obviously. Furthermore, this algorithm is completely software-based and does not require any additional hardware overhead. However, the bad sampling may lead to inaccurate predictions or increased test escape. With the growing complexity of IC, more data will be used for testing, and the sampling will be more accurate. In addition, the reordering method can update the prediction model according to the statistical data, and update the test patterns simultaneously, which shows the method's adaptive ability.

REFERENCES

[1] S. Letchumanan, T. H. H. Tan, Y. P. Gan, and S. L. Wong, "Adaptive test method on production system-level testing (SLT) to optimize test cost, resources and defect parts per million (DPPM)," in *Proc. Int. Symp. VLSI Design, Autom. Test (VLSI-DAT)*, Hsinchu, Taiwan, Apr. 2018, pp. 1–3.
[2] H. Stratigopoulos and C. Streitwieser, "Adaptive test with test escape estimation for mixed-signal ICs," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 37, no. 10, pp. 2125–2138, Oct. 2018.
[3] *International Technology Roadmap for Semiconductors (ITRS)*, Semiconductor Industry Association, San Jose, CA, USA, 2015.
[4] P. Maxwell, "Adaptive test directions," in *Proc. 15th IEEE Eur. Test Symp.*, Praha, Czech Republic, May 2010, pp. 12–16, doi: 10.1109/ETSYM.2010.5512789.
[5] S. Gupta, A. Ansari, S. Feng, and S. Mahlke, "Adaptive online testing for efficient hard fault detection," in *Proc. IEEE Int. Conf. Comput. Des.*, Lake Tahoe, CA, USA, Oct. 2009, pp. 343–349, doi: 10.1109/ICCD.2009.5413132.
[6] N. Li, P. Francis, and B. Robinson, "Static detection of redundant test cases: An initial study," in *Proc. 19th Int. Symp. Softw. Rel. Eng. (ISSRE)*, Seattle, WA, UA, Nov. 2008, pp. 303–304.
[7] C. Xue and R. D. Blanton, "Test-set reordering for improving diagnosability," in *Proc. IEEE 35th VLSI Test Symp. (VTS)*, Las Vegas, NV, USA, Apr. 2017, pp. 1–6, doi: 10.1109/VTS.2017.7928926.
[8] B. Arslan and A. Orailoglu, "Adaptive test optimization through real time learning of test effectiveness," in *Proc. Des. Autom. Test Eur.*, Grenoble, France, Mar. 2011, pp. 1–6, doi: 10.1109/DATE.2011.5763231.
[9] S. C. Ma, P. Franco, and E. J. McCluskey, "An experimental chip to evaluate test techniques experiment results," in *Proc. IEEE Int. Test Conf. (ITC)*, 1995, pp. 663–672.
[10] K. Youn Cho, S. Mitra, and E. J. McCluskey, "Gate exhaustive testing," in *Proc. IEEE Int. Conf. Test*, 2005, pp. 1–7.
[11] F. Ferhani, N. R. Saxena, E. J. McCluskey, and P. Nigh, "How many test patterns are useless?" in *Proc. 26th IEEE VLSI Test Symp.*, Apr. 2008, pp. 23–28.
[12] T. Song, H. Liang, Y. Sun, Z. Huang, M. Yi, X. Fang, and A. Yan, "Novel application of deep learning for adaptive testing based on long short-term memory," in *Proc. IEEE 37th VLSI Test Symp. (VTS)*, Monterey, CA, USA, Apr. 2019, pp. 1–6.
[13] X. Lin, J. Rajski, I. Pomeranz, and S. M. Reddy, "On static test compaction and test pattern ordering for scan designs," in *Proc. Int. Test Conf.*, 2001, pp. 1088–1097.
[14] K. M. Butler and J. Saxena, "An empirical study on the effects of test type ordering on overall test efficiency," in *Proc. Int. Test Conf.*, 2000, pp. 408–416.
[15] R. Daasch, K. Cota, J. McNames, and R. Madge, "In search of the optimum test set," in *Proc. ITC*, 2004, pp. 203–212.
[16] S. Deyati, B. J. Muldrey, and A. Chatterjee, "Adaptive testing of analog/RF circuits using hardware extracted FSM models," in *Proc. IEEE 34th VLSI Test Symp. (VTS)*, Las Vegas, NV, USA, Apr. 2016, pp. 1–6.
[17] V. Katragadda, M. Muthee, A. Gasasira, F. Seelmann, and J.-H. Liao, "Algorithm based adaptive parametric testing for outlier detection and test time reduction," in *Proc. IEEE Int. Conf. Microelectronic Test Struct. (ICMTS)*, Austin, TX, USA, Mar. 2018, pp. 142–146.
[18] F. Brglez, D. Bryan, and K. Kozminski, "Combinational profiles of sequential benchmark circuits," in *Proc. IEEE Int. Symp. Circuits Syst.*, Oct. 1989, pp. 1929–1934.
[19] W. C. Tam, O. Poku, and R. D. Blanton, "Automated failure population creation for validating integrated circuit diagnosis methods," in *Proc. 46th Annu. Design Autom. Conf.*, 2009, pp. 708–713.
[20] R. D. Blanton, J. T. Chen, R. Desineni, K. N. Dwarakanath, W. Maly, and T. J. Vogels, "Fault tuples in diagnosis of deep-submicron circuits," in *Proc. Int. Test Conf.*, 2002, pp. 233–241.
[21] T. L. Quarles, "Analysis of performance and convergence issues for circuit simulation," Ph.D. dissertation, Electr. Eng. Comput. Sci. Dept., Univ. California, Berkeley, CA, USA, 1989. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/1989/1216.html
[22] X. Yu, R. D. Blanton, "Diagnosis-assisted adaptive test," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 9, pp. 1405–1416, Sep. 2012.
[23] W. Zhan and Z. Shao, "Test patterns reordering method based on gamma distribution," *Integration*, vol. 72, pp. 66–71, May 2020.