Efficient Test Strategies for Analog/RF Circuits

by

Ender Yilmaz

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved August 2012 by the
Graduate Supervisory Committee:

Sule Ozev, Chair
Bertan Bakkaloglu
Yu Cao
Jennifer Blain Christen

ARIZONA STATE UNIVERSITY

December 2012

ABSTRACT

Test cost has become a significant portion of device cost and a bottleneck in high volume manufacturing. Increasing integration density and shrinking feature sizes increased test time/cost and reduce observability. Test engineers have to put a tremendous effort in order to maintain test cost within an acceptable budget.

Unfortunately, there is not a single straightforward solution to the problem. Products that are tested have several application domains and distinct customer profiles. Some products are required to operate for long periods of time while others are required to be low cost and optimized for low cost. Multitude of constraints and goals make it impossible to find a single solution that work for all cases. Hence, test development/optimization is typically design/circuit dependent and even process specific. Therefore, test optimization cannot be performed using a single test approach, but necessitates a diversity of approaches.

This works aims at addressing test cost minimization and test quality improvement at various levels. In the first chapter of the work, we investigate pre-silicon strategies, such as design for test and pre-silicon statistical simulation optimization. In the second chapter, we investigate efficient post-silicon test strategies, such as adaptive test, adaptive multi-site test, outlier analysis, and process shift detection/tracking.

ACKNOWLEDGMENTS

I would like to give special thanks to my advisor, Sule Ozev, who guided and supported me throughout my study and made this thesis possible.

Besides my advisor, I would like to thank my thesis committee: Prof. Bertan Bakkaloglu, Prof. Yu Cao, and Jennifer Blain Christen, for their comments and patiance.

My sincere thanks also goes to Anne Meixner and T M Mak of Intel Corporation and Geoff Shofner and LeRoy Winemberg of Freescale Corporation for offering me summer internship and for their insughtful comments and support.

I would like to express my gratituede and thanks to my parents and my sister for their constant support.

Lastly, and most importantly, I wish to thank my wife for her patiance and and support. Without her, I would be completely lost and my effort would be meaningless.

TABLE OF CONTENTS

Page

LIST OF TABLES iv

LIST OF FIGURES v

CHAPTER

1 Introduction 1

1. Design for Test for Built in EVM Measurement 3

2. Statistical Simulation Optimization 6

3. Defect Oriented Test Evaluation 9

4. Adaptive Test 10

2 Pre-silicon Test Strategies 23

1. Design for Test 23

2. Accelerated Statistical Pre-silicon Evaluation/Analysis 51

3. Defect Oriented Test Selection 68

3 Post-Silicon Test Strategies: High Volume Test Optimization 88

1. Per-device Adaptive Test for Analog/RF Circuits 88

2. Adaptive Multi-site Test for Analog/Mixed-signal Circuits 125

3. Adaptive Quality Binning for Analog Circuits 141

4. Multidimensional Outlier Detection 150

5. Efficient Process Shift Detection and Test Re-Alignment 175

REFERENCES 196

LIST OF TABLES

Table Page

1 DFT overhead 44

2 Estimation result and their corresponding standard deviation 64

3 The number of required simulations, and total simulation time for each method. 66

4 Simulation time 82

5 Simulation time savings 83

6 Fail probability and fault coverage table 84

7 Process variation table 116

8 Comparison of test compaction methods for LNA 117

9 Comparison of test compaction methods for production data. 119

10 Comparison of test compaction methods for production data (analog signal circuit) 122

11 Test time and DPPM results for 4 different multi-site configurations 137

12 Test time and DPPM results 140

13 Binning constraints 148

14 DPPM and test time results for data set-1 172

15 DPPM and test time results for data set-2 172

16 DPPM and test time results for data set-3 173

17 Summary of the main features 191

LIST OF FIGURES

Figure Page

1 Design/Test time line . . . 2

2 OFDM transceiver architecture . . . 24

3 Definition of EVM . . . 26

4 IQ signals, rather than IQ symbols are needed for EVM calculation . 27

5 OFDM sequence . . . 28

6 EVM and quantized EVM comparison without DFT . . . 31

7 Constellation enhancement by coding more symbols . . . 35

8 Proposed DFT for built-in EVM measurement . . . 35

9 Analytic quantized EVM estimation . . . 39

10 EVM_Q and EVM vs CER . . . 41

11 Simulation Set-up . . . 45

12 Simulation results for various types of impairments . . . 46

13 Hardware Set-up . . . 48

14 Hardware measurement results . . . 49

15 Estimation variance can be reduced . . . 53

16 Model Based Filtering Flow . . . 56

17 Filtering and weight assigning algorithm . . . 57

18 Model error results in mispredicted circuit behavior . . . 59

19 Introducing guard band prevents the model from making wrong decisions. . . . 61

20 Input space needs to be finely sampled to achieve a good fit . . . 63

21 Experimental circuit: LNA-Mixer . . . 64

Figure Page

22 Skews Model for Process Variation 70

23 Performance parameters in presence of a short between the output node and ground 72

24 WID impact: short between output node and ground 73

25 High-level representation of the Tx driver 74

26 Simulation setup 75

27 Simulation Flow 77

28 Defect pruning critical defect 78

29 Normalized defect severity indicates the defect impact 81

30 Defect severity 86

31 Adaptive testing flow diagram 93

32 Joint probability distribution (JPDF) is represented using multi-dimensional kernels 96

33 Estimated and actual values of Z11 are shown after several update steps for 4 different device instances. 97

34 The devices that fall beyond a pre-determined threshold level are deemed suspicious 99

35 Potentially good devices go through two sanity check steps 100

36 Definition of marginality 102

37 Tightness of a specification is defined as $\frac{\mu}{\sigma}$ 103

38 Training samples that fall close than distance r are combined in order to compact the training sample set. 105

39 KL-distance 107

Figure | Page

40 KL-distance example . . . 110
41 Devices characteristics are analyzed for potential shifts in the process 111
42 Pipe-lined Time Schedule . . . 113
43 Experimental circuit: LNA . . . 116
44 Average Test Time . . . 117
45 Percentage of skipped tests. . . . 118
46 KL-distance of updated curves . . . 120
47 The proposed updating scheme keeps characterization data up-to-date 121
48 Skip histogram . . . 123
49 Flow and estimation engine . . . 127
50 Bulk of the devices require a relatively small number of test . . . 128
51 Compound device . . . 130
52 Estimation of a device parameter using neighbor devices measurements only. . . . 131
53 Multiple devices are tested in parallel to increase the throughput . . 132
54 Initial test list is generated according to the coverage rate . . . 133
55 Neighbor device statistics enable us to use a narrower defect filtering window . . . 134
56 The proposed method and previous work is compared . . . 138
57 Cover based method can be used as a reference for performance comparison . . . 139
58 The dependency can be represented statistically . . . 143

Figure | Page

59 The number of test required to achieve a DPPM level can be represented using a statistical distribution 144

60 Adaptive qualitybinning test flow 145

61 Estimated DPPM is not monotonically non-increasing 146

62 Binning percentages for data set-1 149

63 Vertical dashed lines show the results for binning simulation 150

64 Binning percentages for data set-1 151

65 More than 60% test compaction is achieved for the given constraint sets. 152

66 Outliers can be detected using multiple dimensional analysis 154

67 Profile of the distribution of $D_{\{j\}}(s_i)$ can be modeled parametrically 156

68 Not all parameters bear information 158

69 Adaptive outlier analysis flow 162

70 Outlier analysis flow 165

71 2D visualization of parameters illustrate the operation of the method 168

72 Device distance example 169

73 Test time with respect to testing progress for the second data set . . 171

74 Process shift makes specification parameter\#248 move toward the lower specification limit, increasing probability of failure. 176

75 Wafer-to-wafer process statistics are typically correlated 178

76 Process state is re-learned at every wafer transition 180

77 Wafers are virtually divided into sub-regions and transformation functions are fitted to relate the devices in these sub-region. 181

Figure Page

78 Wafers are partitioned and a transformation function is fitted for each partition separately 183

79 Scatter-plot of the same parameter of two different wafers 193

80 Most of the wafers require as low as 20 samples for re-learning 194

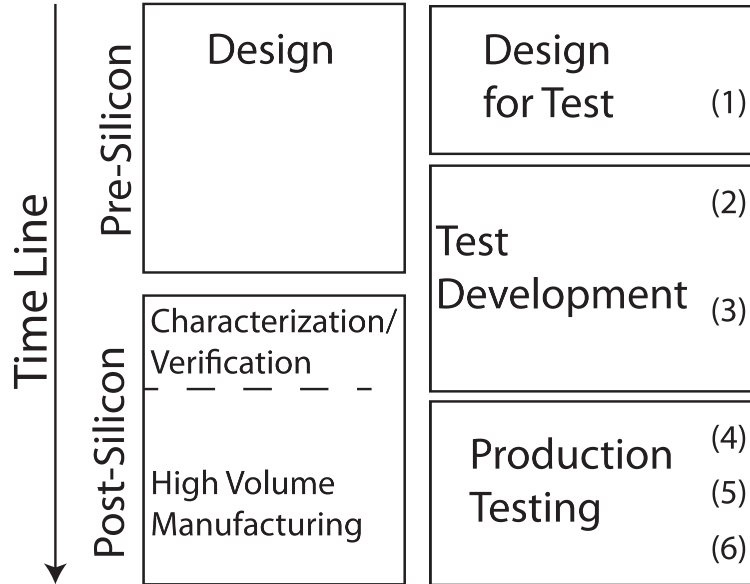
81 Production data results 195

CHAPTER 1

Introduction

Shrinking feature sizes and constantly increasing integration density is imposing an immense pressure on test and testability. Recent studies and future projections showed that test cost has continuously increased and became a significant part of device cost. The trend in continuous feature scaling and integration force test engineers to find more efficient methods. Efficient test methods are investigated at several levels starting from design stage down to production test stage. This is illustrated in Figure 1. Stages of test development and optimization are illustrated with the middle column along with the design stages and product-time-line on the left hand side. In this work, I have concentrated on several aspect of testing and contributed to almost all fields of test. The third column shows the specific section addressed in this work with their respective order in the thesis.

Increasing integration not only increases the number of executed test to cover new functionality but also reduces observability as the number if test pins remained constant while the number of internal nodes increased. Lack of observability lead to the development of design for test methodologies (DfT) in which testing and design steps are integrated. Design for test mainly deals with increasing observability through enabling monitoring internal nodes and providing built-in structures that makes testing process much easier. Although DfT for digital circuits can be automated, automation is very difficult for analog circuits. Analog DfT is typically circuit specific and has to be designed for each circuit. One of the problems addressed in this work is providing a DfT method for error vector magnitude (EVM) measurement to extract a parameter that requires a long test development and test execution time without a DfT approach.



(1) Chapter 2, Section 1: Built in Error Vector Magnitude (EVM) measurement

(2) Chapter 2, Section 2: Statistical simulation optimization

(3) Chapter 2, Section 3: Defect Oriented Test Evaluation

(4) Chapter 3, Section 1: Adaptive Test

(5) Chapter 3, Section 4: Outlier Detection

(6) Chapter 3, Section 5: Process Shift Detection

Fig. 1. Design/Test time line

Test development and design are lengthy processes and are desired to be performed in parallel in order to shorten design and verification cycle. Test development is performed in early post silicon stage once silicon data becomes available. The data obtained from the first batches are used to characterize the manufactured devices and design an optimal test list. However, waiting until the post-silicon stage is not an efficient strategy. Moreover, some of the analysis require extensive data that exacerbates the waiting period. Therefore, test development and yield/defect oriented design evaluations are shifted to overlap with

design stage. This corresponds to the test development process in pre-silicon stage in Figure 1.

In this stage design is evaluated using simulation and prior knowledge of the process to estimate critical parameters such as yield and test escape rate for a given set of tests. Estimation of these parameters is crucial to prevent yield loss and generation of an optimal test list as diversity of devices increase with new technology nodes. However, this step is computationally intensive. Despite of the development in computer technology, simulation times are projected to be several hundreds of years even for a medium scale circuit. In section 2.2, simulation bottleneck of pre-silicon evaluation is addressed and an efficient method is proposed that greatly reduces simulation overhead.

There are two main approaches in test selection: specification based test selection and defect based test selection. The former approach uses specification parameters that are defined according to a standard or requested by customers for test development and optimizes the test list to reduce test time. While, the latter approach deals with optimizing the test list with respect to defects. In this work, we address defect based test selection method in pre-silicon stage (section 2.3) proposing efficient methods that are demonstrated using industrial circuits. We address specification based test optimization in post-silicon, production testing stage.

1. Design for Test for Built in EVM Measurement

Due to its soft capacity and frequency diversity, orthogonal frequency division multiplexing (OFDM) has become the prevailing modulation scheme for modern RF transceiver systems. Modulation accuracy, specified in terms of error vector magnitude (EVM), is one of the most important transmitter parameter specified for OFDM transceivers. EVM encapsulates many non-idealities of the transmitter, including inter symbol interference,

mismatches, non-linearity, phase noise and spurs, and carrier leakage [Gu(2005)]. It is typically one of the few parameters specified for the transmitter and needs to be guaranteed.

Recently, several researchers have proposed techniques for simplified EVM measurements. An optimization method for EVM measurement that reduces the overall test time for EVM characterization is presented in [Acar *et al.*(2008)]. The goal is to limit the number of symbols that need to be transmitted/received within one frame as dictated by the normal operation mode. The complete receive operation is duplicated and conducted at the tester using a golden receiver on the load board. In [Natarajan *et al.*(2008)], the authors propose an EVM measurement method that requires less number of measurements by increasing the sensitivity of EVM to transceiver system noise. A computationally efficient EVM measurement method for phase-only modulation schemes is presented in [Helfenstein *et al.*(2005)]. EVM test sequence is reduced in [Acar *et al.*(2006)] by selecting sensitive corner cases into the test vector. Hence, test sequence required to achieve the desired confidence level is shrinked.

Another trend in reducing the EVM test cost is using alternate testing methods. Instead of modulated signals, multi-tone input signals are employed in [Halder *et al.*(2008)] to estimate the EVM. Resulting waveforms are transformed into EVM through nonlinear regression. EVM is estimated by utilizing null carriers to calculate system noise in [Senguttuvan *et al.*(2008)]. Thus, test time required to gather adequate information from the device is reduced. A multi-tone based method is presented in [Bhattacharya *et al.*(2005b)] for Ultra Wide Bandwidth (UWB) systems.

While there has been significant progress in reducing the overhead of EVM measurements, important challenges still remain, especially when it comes to multi-site testing. Characterization of EVM requires sending multiple signals spread over multiple frames and

conducting the overall receive operation including the channel characterization and synchronization. The I and Q signals need to captured and analyzed at the tester requiring access to these signals. Moreover, the complete receive operation needs to be duplicated by the test engineer and incorporated into the test program. This operation requires complex procedures to synchronize the transmitted and received frames and to estimate the channel characteristics. These steps complicate the test development process and may result in long test development times.

The need to analyze the IQ signals, rather than the bit pattern, also puts a high burden on the tester. The IQ signals are typically digitized with a resolution of 8 bits [Saponara *et al.*(2008)]. This means that measuring EVM requires 16 bits of digital bandwidth between the tester and the device under test (DUT) just for the analog signals. As a result, multi-site operation becomes harder.

Perhaps a more important burden is placed on the computational resources of the tester [Srinivasan *et al.*(2008)]. Some of the steps of the receive operation require compute-intensive algorithms, sometimes requiring multiple passes through the IQ samples [Acar *et al.*(2008)]. This computational overhead imposes a bottleneck in increasing the multi-site ratio during EVM testing. A potential solution to this problem has been proposed in [Srinivasan *et al.*(2008)].

We take a different approach and propose to place design-for-testability techniques into the digital base-band subsystem of the device to enable low-overhead built-in EVM testing. The intuition is that the receiver and the transmitter of a transceiver system are designed to perform complementary operations in the base-band and all the complex steps of the receive operation are already implemented for this particular platform. To avoid having to take IQ signals out of the chip, we propose a DFT approach that enables us to measure

EVM through only the decoded bit pattern, greatly reducing the test development time and computational burden on the tester. We propose an innovative method for measuring EVM on-chip with very little DFT overhead. To this effect, we exploit the symbol coding scheme and the fact that displaced symbols present deterministic bit flip characteristics in the coded bit pattern. The DFT approach presents less than 0.05% overhead for the transceiver excluding any DSP or controller area.

We present an analytical model that relates the EVM measurement accuracy to the number of enhanced constellation points and a detailed analysis of the overhead imposed by the proposed DFT scheme. We conduct experiments using both a MATLAB simulation platform and through hardware measurements and observe that we can measure the EVM with less than 1% error.

2. Statistical Simulation Optimization

Customers demand estimation of test quality metrics. The most prominently used quality metric is Defective Parts Per Million (DPPM), which is defined as the number of defective devices shipped to customers out of a million shipped devices. DPPM is generally estimated based on available test set and failure modes. Defect escapes typically happen due to compaction of large test lists that are not feasible to apply in high volume manufacturing.

The most widely used specification-based compaction method is using a minimum set cover approach, which determines a subset of the original test list based on fall-out patterns. This compaction technique can be implemented using the Integer Linear Programming (ILP) approach [Drineas and Makris(2003)], heuristic methods [Milor(1998)], or machine learning approaches, such as SVM (Support Vector Machine) [Biswas *et al.*(2005)]. Test compaction methods reduce testing time at a cost of increasing DPPM; defect escapes occur due to skipping the tests that are unlikely to fail but have a nonzero fail probability.

Another form of test compaction is to use a set of alternate measurements [Haider and Chatterjee(2005),Akbay and Chatterjee(2004)] or using dedicated built-in-self-test circuitry that indirectly decides on the pass/fail status of the device [Hafed and Roberts(2000), Petlenkov *et al.*(2008)]. For these techniques, defect escapes occur due to imperfections in the modeling/mapping functions.

Clearly, DPPM of any given test set needs to be estimated to ensure high test quality during the test development phase. However, estimation of DPPM is considerably difficult since it is a complex statistical parameter. Typically, such statistical parameters are estimated using Monte Carlo (MC) simulations. However, using MC simulations for accurate DPPM estimation is unaffordable due to the need to simulate a large number of samples.

Several approaches have addressed the need to reduce the computational burden of MC simulations. The main concern in estimating a statistical parameter is achieving the desired precision in a small number of simulations. Precision of an unbiased estimation technique can be measured using the variation of the estimated parameter. Therefore, the goal of an efficient statistical parameter estimator is to yield lower variation for the same number of simulations or same variation for less number of simulations. A re-sampling based approach is proposed in [Stratigopoulos *et al.*(2009b)], where a small sample set of devices is exhaustively simulated and the result is used to synthetically generate a larger device set. While this method is quite efficient and provides an excellent match for the bulk of the samples, it suffers from the lack of samples at the boundaries of the process space. Thus, for DPPM estimation, the variation can be large. Quasi Monte Carlo (QMC) sampling proposed in [Singhee and Rutenbar(2010)] uses low discrepancy sequences to sample the input space. QMC ensures even coverage of the input space but it still generates a large number of samples. Experiment design techniques, such as Taguchi based methods [D'Errico

and Zaino(1988), Taguchi(1978)] can reduce variation in the results and ensure wide-range coverage of the input space but are generally unaffordable for for high dimensional spaces.

In [Singhee and Rutenbar(2008)], an Extreme Value Theory (EVT) based rare event simulation method has been proposed to estimate the tail distribution of individual specification parameters. Similarly, [Stratigopoulos and Mir(2010)] employs an EVT based approach to estimate yield loss and test escape level for one of the tests that is replaced with an alternative test. But, these two methods cannot be applied to DPPM estimation for multiple dimensional output space, which requires multidimensional tail distribution estimation.

Another way of improving efficiency in DPPM estimation is importance sampling. The goal of importance sampling is to sample only instances that are relevant in order to improve efficiency. In [Kanj *et al.*(2006)], the importance sampling concept has been applied for SRAM circuits, where process parameters are virtually shifted to sample from the region of interest. Unfortunately, importance sampling requires knowledge of the importance regions in the process parameter space. For analog circuits with multiple specifications and complex relations between process and specification parameters, finding such importance regions is not trivial.

In this work, we propose an intuitive and innovative method by combining the concepts of importance sampling and model fitting to increase the efficiency of MC simulations. Basically, we use a model estimator of the circuit behavior to determine the importance region. We avoid the need for extremely accurate model generation by using adaptive guard banding and robust model generation. In order to effectively achieve importance sampling without the necessity of knowing the importance regions in the process space, we use the model as a filter for an MC simulator. We evaluate generated MC samples by the model

and simulate only those that have a potential impact on the DPPM parameter, thereby greatly reducing the simulation effort. Hence, we do not avoid the simulation step, but lift the heavy burden on the simulator by making informed decisions on the MC samples through a simple model, and let the simulator work on a refined small number of device instances.

We demonstrate our approach for a receiver front-end circuit under process variations. Hence the failure model that we use is based on process variability. However, our approach can be used in conjunction with any failure model to evaluate DPPM of a given test set. This information is extremely useful during test development to make informed decisions on the test list.

3. Defect Oriented Test Evaluation

Today's assessment of analog test coverage consists of ad-hoc checklists of analog tests and exercising all the analog circuitry in some manner. We have no knowledge as to the importance of one test over another. As analog designs change to adapt to process constraints and product requirements, evaluating test coverage during product definition and pre-silicon validation adds value to the design and test process.

A very promising method, analog fault modeling (AFM), which enables such coverage assessment, has been proposed nearly two decades ago. Early work on AFM focused on parametric defects and circuit level defects. [Milor and Visvanathan(1989)] proposed an approach with process variation; while [Meixner and Maly(1991)] investigated circuit level defects. Parametric faults are typically simulated with out of tolerance deviations [Soma(1991),Bishop and Ivanov(1995)], while open and short defects are simulated via injecting respectively a large and small resistance [Azais *et al.*(2003),Chang *et al.*(2002),Sebeke *et al.*(1995), Stratigopoulos *et al.*(2009a)]. Early work on defect modeling did not

include the masking effect of process variation [Nagi and Abraham(1992), Voorakaranam *et al.*(1997)], which is becoming increasingly prominent with more advanced processes. Process variation is incorporated in AFM [Milor and Sangiovanni-Vincentelli(1994), Chao *et al.*(1997)] at a cost of increasing computational complexity. Although AFM provides great insight in defective behavior and test coverage, it has remained an academic research topic due to its extensive computational requirements.

In this work, we provide a feasible implementation methodology for defect-oriented simulations that substantially reduces the computation requirements. We exploit the hierarchical structure of process variation to split the simulation process in multiple steps and apply a pruning algorithm to eliminate unnecessary steps. In the first step we analyze the impact of die-to-die variations. This first hand analysis gives us defects that have an effect on performance. In the second step, we include within die (WID) variation for the defects that have a possibility of resulting in specification violation.

By demonstrating the AFM approach on a commercial device, we show that fault simulation is both feasible and provides invaluable information for test quality optimization and yield improvement. The results of this defect analysis can be used for many purposes. In this work, we analyze the coverage of the manufacturing tests based on this defect analysis. We also present a methodology to identify sensitive nodes in the circuit, so that they can be targeted in the layout step for yield improvement.

4. Adaptive Test

Performance of mixed-signal and RF circuits is typically defined by a large number of diverse specifications. Some of these specifications are highly correlated, suggesting that measuring all of them during production test is redundant and results in unacceptably long test times. As a result, the common approach in the industry has been to collect

statistical information on a small number of devices during characterization/production ramp-up phase and to use this information to compact the test set to reasonable levels. These devices serve as a training set to determine which tests to apply. Several approaches have been proposed to use the learned information to find an efficient way of eliminating tests. However, test compaction level and achievable test quality of these methods are extremely limited. Constant pressure for test cost reduction necessitates replacement of these inefficient methods with more efficient ones that can offer superior performance.We provide a detailed discussion on the prior work in the next section.

With increasing process variations, the statistical diversity of devices also increases. In most cases, specifications cover a wide range of distributions and there are devices that fall further off from the nominal space. This is particularly problematic for marginal devices that are close to the specification limits. Test compaction makes test quality metrics very sensitive to the accuracy of correlations between the specification parameters, especially considering that the correlations are subject to change with process shift. Even a small error may move a device from the acceptable region to the unacceptable region and vise verse. Devoting the same test resources to each device (i.e. applying the same test set) is therefore not an efficient use of these resources. Marginal devices should be tested more extensively whereas devices that conform to the learned statistical information can be passed more easily. Such a test flow that tailors the test set for each device based on where it falls in the process space is adaptive on a per-device basis and can potentially provide the best test time-test quality trade-off. Recently, we have proposed several approaches with this philosophy [Yilmaz and Ozev(2008), Yilmaz and Ozev(2009a)]. In our earlier work, we use a simple correlator to extract this information and show the potential of improvement. However, more elaborate and efficient methods are needed to exploit the whole potential.

Another important issue in test compaction is monitoring process shift and efficient re-characterization to maintain test quality level. Unfortunately process statistics are not stationary, but shift over time. Test compaction methods typically use a single snapshot of the process in their analysis and do not adapt to the changes. However, process statistics are subject to change and test compaction methods have to adapt in order to maintain characterization data representative. Failing to do so may increase misclassification rate and greatly degrade the performance of the used test compaction method. To remedy this problem, some production test flows incorporate mechanisms to check the validity of learned collective statistical information periodically by subjecting a set of randomly selected devices to more extensive testing. Several commercial vendors provide tools to systematically adapt the test set with respect to changes in the process profiles, although algorithms and methodologies that are used are typically not published.

In Section 3.1, we present a methodology for per-device adaptive test flow using a kernel-based estimator. We provide techniques to pre-select a number of tests to collect essential information on each DUT. Once these tests are applied, the results are processed to determine the next phase of testing. This process is repeated until all tests are either applied or marked as *pass* without direct measurement. We also provide differential entropy based process tracking/update mechanism to update changing process information. The characterization data is continuously monitored for potential process shifts and updated as necessary to maintain its representativeness. Re-learning rate is adjusted to catch-up with the speed of change in the shifts. In order to ensure that defective devices are screened out as much as possible, we provide a mechanism to check whether each device conforms to the expected behavior. Suspicious devices are identified and are subject to more extensive testing even if they *pass* all measured parameters. We have applied this adaptive

technique both in simulations to a low noise amplifier (LNA) and to production data of two diverse mixed-signal/analog devices. All three results indicate that the proposed adaptive techniques achieve the lowest DPPM values when compared with static techniques.

4.1. ***Multi-Site Adaptive Test***

Yet another common industry approach to test time reduction is multi-site testing where more than one device is tested at once. Multi-site testing is particularly attractive for analog circuits since analog tests do not require a high pin count. A natural extension to adaptive test is thus to apply it in a multi-site environment.

However, since multi-site testing requires placement and removal of all DUTs on the load board at the same time, direct application of per-device adaptive test is not feasible [Yilmaz and Ozev(2010)]. While static compaction methods easily lend themselves to linear test time reduction with multi-site testing, adaptive test approaches suffer from sub-linear test time reduction since all sites must finish their tests before the devices can be removed from the tester. This invariably results in test times that approach the test time of most marginal devices, which are tested more extensively.

In Section 3.2, we address this problem and propose a solution to multi-site adaptive testing to reap the benefits of multi-site testing and adaptive testing at the same time. Our method is based on the observation that devices that are tested in parallel generally originate from the same wafer and are closely related in terms of their statistical characteristics. We exploit this property in two ways: (1) we use a compound device approach to utilize the information from all sites for each DUT, and (2) we use the common tested parameters of all DUTs to screen for defects more efficiently. Experimental results based on production data from two diverse mixed-signal circuits indicate that despite the constraints imposed by multi-site testing, our approach helps scale the adaptive test time linearly and attain the highest test quality compared with prior work.

4.2. *Adaptive Quality Binning*

Device cost can be minimized if only the requested amount of the produced devices barely meet the specified performance limits. In other words, if there are multiple quality bins, we would like to produce devices that split in these bins at a desired ratio. However, this cannot be achieved with traditional test methods. Optimization of quality binning can be performed production test phase. Device cost can be reduced if test resources are distributed with respect to each device's quality.

One approach to attack this optimization problem would be to follow a sequential approach: randomly partition devices according to a given proportion criteria, and then apply the adaptive testing procedure described in the previous sections. Although simple and straightforward, this approach does not guarantee a global minimum because the sequential approach assumes the DPPM and time to be independent. In fact, they are highly correlated and this can be used to optimize the problem toward a global solution. Instead of using two consecutive steps, we attack the problem is a single step by exploiting the strong correlation between DPPM and test time. We use per-device measurement data to estimate the expected defective escape probability and test time to achieve that of each device jointly. Hence, we can estimate test time for specific DPPM ranges and bin the devices not randomly but according to their behavior. If a device does not show high quality behavior, it can be put into the lower quality bin and end testing sooner. However, if a device is expected to have a high quality profile, it can be tested further until the desired quality level is achieved. This approach yields lower expected test time since high quality devices require less number of tests.

In this work, we aim at adaptively selecting a test list for each device considering quality requirements and devoting only required amount of test resources to minimize the

test cost. We speed bin devices according to a well known test quality metric, defective parts per million (DPPM).

4.3. *Test Quality Oriented Outlier Analysis*

Circuits fail either due to process induced variations or due to inherent defects. Process-induced failures can be tracked more easily using collective information since most test compaction techniques use the collective distribution of data to make test decisions [Stratigopoulos *et al.*(2007), Chen and Orailoglu(2008)]. Circuits containing defects however, behave in a random manner, invalidating learned information, and making their detection using a reduced set less likely.

This random behavior of defective circuits can be detected through outlier analysis, where the goal is to identify devices that behave differently from the bulk of devices. In the context of testing, outlier analysis has many applications. It can be used in conjunction with a test selection technique to reduce DPPM, or with alternate test techniques [Bhattacharya *et al.*(2005a)], as a defect filter. In some domains, such as the automotive domain, outliers can be outright rejected regardless of whether they pass or fail the specifications.

Four important challenges must be overcome for efficient outlier analysis. First, setting outlier boundaries is typically difficult due to high process variations. Second, for analog circuits outlier analysis needs to be conducted in multiple dimensions that can be easily in the order of the hundreds. Third, process shifts may render pre-determined limits invalid. Fourth, inclusion of parameters that have no distinguishing capability introduces uncertainty into the decision mechanism and diminishes the efficiency. Moreover, these parameters depend on the population and cannot be statically determined. Various outlier

analysis methods have been proposed in the context of testing, which address one or more of the above-mentioned challenges.

Several methods have been proposed for variance reduction in outlier analysis in the context of Integrated Circuit Quiescent Current (IDDQ) testing [Maxwell *et al.*(2000), Turakhia *et al.*(2005)] using current ratios, differential currents, or neighborhood information.

In [Fang *et al.*(2006)], an adaptive scheme is proposed, which works in one dimension. In [Yilmaz and Ozev(2009b)], a two-dimensional static outlier analysis is used. In [Cerioli(2009), Filzmoser *et al.*(2005), Papadimitriou *et al.*(2003), Pena and Prieto(2001)] present multidimensional outlier analysis methods are proposed. However, none of the above-mentioned techniques address all issues at once and/or cannot be easily extended for this purpose.

In this work, we present an adaptive multidimensional outlier analysis method that combines multiple information rich parameters and adaptively incorporates the evolving statistics of device parameters. We model multi-dimensional data using kernel-based density estimation. In this way, we can effectively and accurately keep track of non-linear dependencies. We propose a method to determine a subset of conducted measurements for outlier analysis to reduce the uncertainty induced by parameters that provide no distinguishing information. We continuously update our profiles to keep track of shifts in the process, which changes the outlier limits adaptively.

The proposed method can be integrated with an adaptive test framework. We use our earlier work [Yilmaz and Ozev(2010)] for this purpose and demonstrate how outlier analysis can be used to reduce test time while attaining high test quality.

4.4. ***Efficient Process Characterization***

Production test is one of the major contributors of the product cost. Typically, a large number of specifications are tested to guarantee device functionality and this makes production test a time consuming process. Test time is typically shortened using various methods. Since most of the specification parameters are correlated, specification-to-specification dependencies are used to shorten the test list by eliminating some of the correlated tests. This problem is known as analog test compaction.

Various analog test compaction methods have been investigated extensively and some form of test compaction is generally used in the industry. In the next section, we will explain some of these approaches in detail. Generally, information that is learned from a representative set of devices (referred to as characterization data), typically obtained during production ramp-up, is used to improve the test time or test cost by reducing the test list or by finding simpler tests to replace the ones on the test list.

All the statistical test compaction methods, including the set cover based methods, rely on the accuracy and the representativeness of the characterization data set. Incorrect characterization data may result in high misclassification rates and therefore degradation in test quality. Unfortunately, regardless of how much information is collected on the statistical characteristics of the devices initially, this information eventually becomes invalid, at least partially, due to changes in the underlying process parameters. In order to maintain a high test quality level, characterization data should be maintained up-to-date.

Process shifts may be due to intentional adjustment by process engineers or an unexpected behavior that alters the process statistics. Even if the changes in process level parameters (i.e, doping density, temperature,...) are known, the effect is typically not easily predictable in terms of specification domain parameters. This necessitates re-learning the process information to avoid any misclassification. Failing to detect a shift and take action by updating the test engine in time may result in an increase in the misclassification rate. However, re-characterization when the shift is not significant results in excessive overhead. Therefore, it is important to use an optimized process shift detection and re-learning method.

In this work, we present an efficient process tracking scheme in the the specification parameter domain to keep characterization information valid with a very small number of measurements. We exploit wafer-to-wafer correlations to minimize the re-learning effort. We re-use available characterization information (outdated) and transform it to estimate the process statistics of the target wafer that is being tested. We show that only a very small number of samples from the target wafer are sufficient for such a transformation. The proposed method complements test compaction efforts and can be integrated with any such technique (e.g. [Akbay and Chatterjee(2007), Biswas and Blanton(2006), Yilmaz and Ozev(2010), Stratigopoulos *et al.*(2007)]).

We use a *2-detect set-cover* method for test list generation. We update test list incrementally for each wafer using the transformed characterization data. Thus, test list is tailored with respect to statistics of the WUT. Since wafer-to-wafer shifts are not always extensive, most of the existing information on which tests to include can be re-used. Thus, most of the time, there is little computational overhead. When large-scale shifts do occur, we deem the existing information invalid, and conduct a more comprehensive re-learning

step. Once a test list is generated, it is applied to all devices on that wafer. Executing a common test list enables us to easily adopt multi-site testing capability and therefore reduce test time.

4.5. *Prior Work*

Over the past several decades, many approaches have been proposed to make the test compaction process systematic and efficient. The most simplistic view of test compaction arises from observation of fall-out patterns of individual tests. In this sense, test compaction can be viewed as a set-cover problem wherein the goal is to select a minimal subset of existing tests such that all fall-out cases in the training set are identified. Various algorithms have been developed to efficiently identify this minimal cover since the set-cover problem itself is NP-hard [Cormen *et al.*(1997)]. ILP-based formulation [Drineas and Makris(2003), Stratigopoulos *et al.*(2007)] has been shown to produce near-optimal covers [Yilmaz and Ozev(2009a)]. Heuristic approaches [Milor(1998), Biswas and Blanton(2006), Biswas and Blanton(2008)] have also been proposed to identify a covering test set efficiently when the full test set is large and ILP-formulation is no longer computationally feasible. Once a test set is identified, test set re-ordering [Huss and Gyurcsik(1991), Jiang and Vinnakota(1999)] is proposed to further reduce the amortized test time by applying the most likely to fail tests first.

Machine learning methods have also been used to achieve better generalization and more accurate modeling. The use of artificial neural networks (ANN) for this purpose is presented in [Biswas *et al.*(2005)] to improve classification performance. Similar methods for statistical modeling based on ANNs or kernel density estimation (KDE) have been proposed in [Stratigopoulos and Makris(2005), Stratigopoulos *et al.*(2007)] and [Yilmaz and

Ozev(2010)]. These methods essentially divide the high dimensional specification parameter space using complex boundaries to achieve an optimized test set with better characterization performance compared to the set-cover method. An adaptive test method is proposed in [Yilmaz and Ozev(2010)] to generate a device specific separation region, which is updated for each device, to achieve a much finer boundary and therefore superior classification performance. Due to device-level adaptation, this method cannot be easily adopted to multi-site testing.

Alternate test methods use non-specification to specification parameter correlations [Bhattacharya *et al.*(2005a), Kupp *et al.*(2009)]. These approaches essentially combine the benefits of low-cost and short test time overhead of non-specification tests and map them onto specification domain using nonlinear transformation functions.

Process shift adaptation for analog circuits is a relatively unexplored domain. One of the first efforts in achieving process adaptation using a lot-to-lot update mechanism is presented in [Benner and Boroffice(2001)]. The author proposes to re-learn the test statistics and re-optimize the test list every time a new lot is encountered. In the next sections, we will show that such pre-determined update frequencies can lead to either excessive overhead or poor test quality. In [Chen and Orailoglu(2008)], test compaction is adapted to local process changes using the process capability factor (CPK). The general strategy of process adaptation methods is to use the CPK information to update the test list by including marginal tests or by initiating a re-learning step. The re-learning step involves applying a full test suite to a large set of devices as in [Chen and Orailoglu(2008)]. Although both of these strategies offer adaptation to process shifts, their test time reduction capability is limited. Adding marginal tests increases on-line test time since specification-to-specification correlations are not used for test compaction. However, re-learning at a rate faster than

the process shift speed (over learning) increases test time overhead. A continuous learning scheme is adopted in [Gotkhindikar *et al.*(2011)] to update failure rate information and achieve die-level adaptation. Test compaction rate of this method is very limited for high yield processes.

In [Kupp *et al.*(2011)], incorporation of scribe-line information is investigated to reduce test time overhead. Scribe-line measurements are correlated to the specification domain parameters and can be used to reduce test time through mapping scribe-line data to specification domain parameters. However, generation of a function to do the mapping requires serious computational effort. Thus, it would be prudent to do such re-learning as rarely as possible.

CHAPTER 2

Pre-silicon Test Strategies

1. Design for Test

In this section, we present a technique to enable accurate built-in measurement of EVM for OFDM transceivers. This measurement technique only relies on the decoded bit pattern, and does not require any additional test equipment. In order to accurately predict EVM without using analog signal analysis, we intentionally code more symbols into the bit pattern in test mode, which enables the decoding of IQ signals in finer granularity. We present an innovative DFT technique to measure EVM on-chip with very little overhead. We also provide an analytical framework to determine how the DFT technique needs to be implemented. Experimental results using MATLAB simulations and hardware measurements confirm the accuracy of the proposed technique.

1.1. *OFDM Transceivers and EVM Measurement*

In this work, we focus on OFDM transceivers. A generic OFDM transceiver is shown in Figure 2. The input bit pattern is enhanced by convolutional encoding, error correction and interleaving. The S/P block takes 2-6 bits of data (depending on mode) and converts them into IQ symbols. The symbol mapping block codes the bit pattern into frequency domain I and Q signals, typically 8-bits for each [Saponara *et al.*(2008)]. These I and Q signals are passed through the IFFT block to generate the time domain signals, converted into the analog domain and transferred into the RF subsystem. The receiver performs the complementary operation.

Baseband signals are distorted by many non-idealities during the transmit/receive process. For signals with appreciable power, the distortion stems from the transmitter's

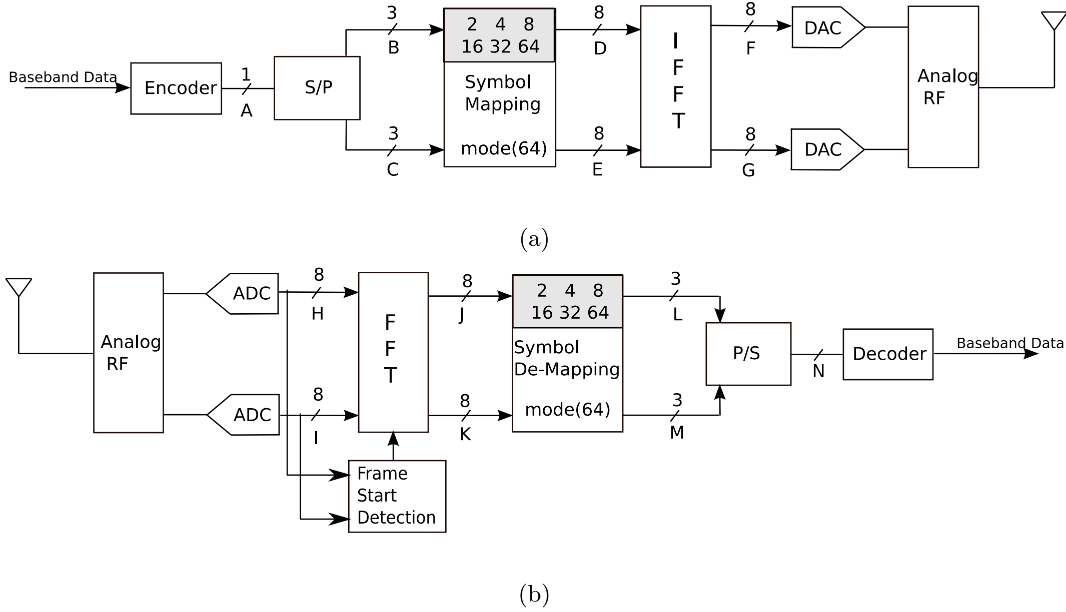


Fig. 2. OFDM transceiver architecture

non-idealities, such as inter-symbol-interference (ISI), carrier leakage, DC offsets, IQ mismatches, and non-linearity [Gu(2005)]. Error vector magnitude is one of the metrics defined for modulation accuracy and it is specified in standards using the OFDM scheme. EVM basically defines the amount of displacement of the received symbol from its ideal location in the complex IQ plane. As an example, Figure 3(a) illustrates the constellation diagram of QAM16. Constellation points shown as circles are the ideal positions of the symbol constellations. Points that are shown by plus (+) signs represent symbols that are affected due to unwanted distortion or noise mechanisms. Impairments or induced noise result in changes in the constellation points. Thus, the quality of the received signal can be estimated by comparing the received symbol constellations with their ideal locations.

Figure 3(b) zooms into the area around one of these symbols. The vector between the ideal symbol location and the received symbol location is the error vector, and the magnitude of that is the error vector magnitude. The instantaneous EVM value is affected

greatly by thermal noise and measurement errors. This randomness can be rather high so the instantaneous EVM value is typically of little use. A more useful metric is the root-mean-square of the EVM over a number of symbols. In fact, many standards define the EVM specification based on its RMS value:

$$EVM = \frac{\sum_{i=1}^{N_f} \sqrt{\frac{\sum_{j=1}^{L_P}(\sum_{k=1}^{N_C}(I_{i,j,k}-I_{0i,j,k})^2+(Q_{i,j,k}-Q_{0i,j,k})^2)}{L_P N_C P_0}}}{N_f}$$

Where, N_f is the number of frames, L_P is the length of the packet, N_C is the number of carriers, P_0 is the power of constellation. Ideal position of the symbols I and Q is give as I_0 and Q_0 respectively. As an example, IEEE 802.11 [iee(1999)] standard defines EVM as the RMS of EVM values for 320 randomly selected symbols.

While the EVM is specified for OFDM transceivers, there is an inherent randomness in the EVM parameter due to the randomness of the input pattern even when the measurement errors and other factors are taken out of the equation. It has been shown in [Acar *et al.*(2008)] that measurement of EVM with two random symbol sequences, each being 320 symbols long, can have as much as 1% deviation. One has to take this uncertainly into account when designing a test approach using EVM.

1.2. *EVM Calculation*

Calculation of EVM requires comparison of the actual received signal with the ideal received signal in the complex IQ plane. As such, one needs to generate the received IQ signals, without mapping it onto the actual bit pattern.

As an example, consider the following scenario: in a QAM-16 scheme, the mapping between the bit pattern and the symbol locations (illustrated as the circles) in the constellation diagram are given in Figure 4. We want to compute the EVM associated with the

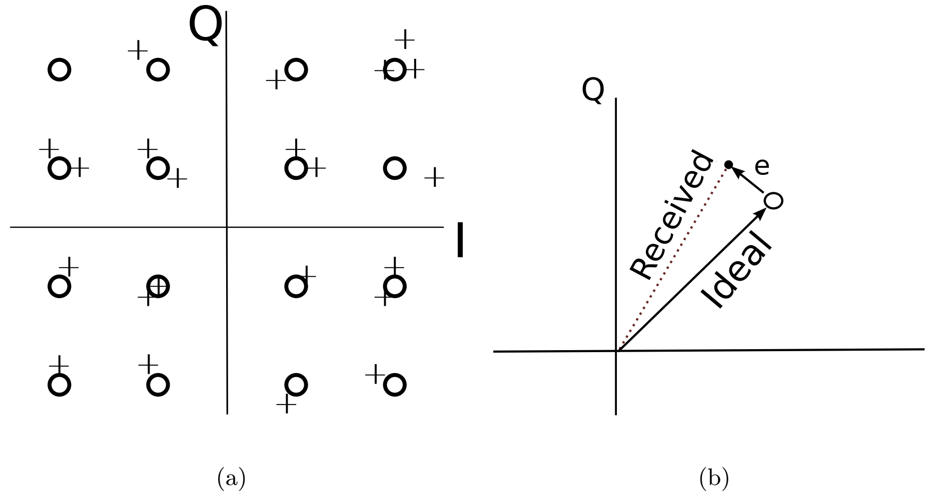


Fig. 3. (a) Constellation diagram for QAM16, (b) Definition of EVM

bit pattern (1010), which is translated into the complex base-band signal as (1.5+j0.5). We input this signal into the transmitter under test three times. Suppose on the receiver side, the received base-band signals correspond to (1.8+j0.3), (1.2+j0.8), and (1.5+j0.1), illustrated as "x" in Figure 4. The resulting RMS EVM from these symbols is 14%. If the bit pattern is decoded however, we see that all three received symbols fall within the decision boundary of the transmitted symbol, thus there will be no errors. This simple example demonstrates the need to process the IQ data before it is decoded into the bit pattern to calculate the EVM. In section 1.8.1, we show simulation results that qualitatively points this shortcoming.

The steps involved in EVM computation can be summarized as follows:

- Detection of the start of the frame
- Estimation of the channel
- Normalizing the received signal

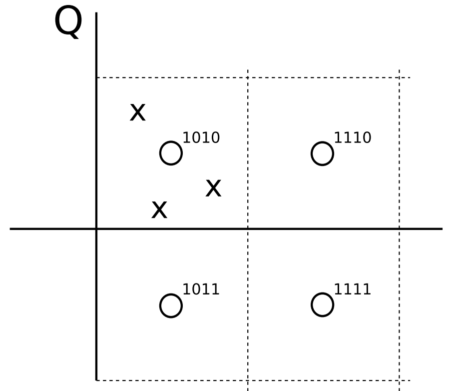


Fig. 4. IQ signals, rather than IQ symbols are needed for EVM calculation

- Mapping
- Calculation the displacement of the received constellation points
- Computation of the RMS average of all errors

Since OFDM systems are coherent, the raw received signal has to be synchronized with the transmitted signal. There is typically no standard-determined procedure. Typically, signal processing methods are used to capture the start of the frame by using short and long training sequences with custom signal processing methods [Liu(2003), Hanzo *et al.*(2003)]. Illustration of an OFDM sequence is given in Figure 5. One method to locate the starting point is to use the auto-correlation function of the short signals contained in the header of the packet to coarsely determine and confine the search region [Acar *et al.*(2008)]. Achievable phase resolution of this method is limited due to noise. Therefore, additional steps are used to exactly point the start of the frame. Fine tuning is achieved through utilizing cyclic redundancy of long signals.

The next step is estimation of the channel. Estimation of the channel is done using the long symbols of the header, as has been suggested in the standard. Long symbols contain predefined values of the same power, -1 and +1, with zero mean. If the channel is frequency selective, long symbols obtained at the receiver side are of non-even energy. Since each carrier is modulated with these zero-mean constant-energy signals, channel can be estimated at all points of the carriers. Estimation of the channel is followed by normalizing the signal using this channel model. Thus, effect of the channel on the received signal is removed.

The next step is mapping the symbols obtained in the previous step onto the constellation diagram. Although channel is estimated in order to remove its effect, it is not possible to recover the transmitted symbols with no error. Noise and system non-idealities that are explained in detail in the following sections, prevent error free detection of the symbols. EVM is basically a measure of the amount of displacement of the received symbol from its ideal constellation point. RMS value of the errors of all received symbols are averaged and normalized using the total constellation power in order to obtain the EVM value.

1.3. *Challenges in Measuring EVM*

One of the challenges in measuring EVM is the test development time. A receiver/transmitter requires a complementary functioning test equipment to measure the functionality of the device. For instance, testing of a transmitter requires down-converting equipment and ATE (Automatic Test Equipment) programmed to decode the modulated

Fig. 5. OFDM sequence

signal and analyze the distortion of the base-band signal. Calculating the EVM performance of a device requires either generation or analysis of RF domain signals. If all the RF/analog signal generation is done by the tester, multi-site operation would not be possible since testers typically have very limited parallelization capability when it comes to RF/analog signal analysis. This bottleneck can be eliminated by employing fully characterized down-converters and up-converters [Srinivasan *et al.*(2008), Acar *et al.*(2008)].

Another difficulty of calculating EVM is the computational burden required to process and analyze the test signals. The first step of EVM calculation is the determination of the frame start. This step typically requires computationally intensive operations [Liu(2003), Hanzo *et al.*(2003)]. While there is no standard algorithm for this process, the common thread in various proposed approaches is that they require multiple passes through the received data. In [Srinivasan *et al.*(2008)], the authors aim at solving this computational bottleneck problem by including DSPs on the load board. In this way, complex operations of the receiver base-band can be offloaded to these DSPs, relieving the tester. While very effective, such a test development effort requires extensive knowledge of modulation/demodulation schemes as well as programming and debugging of the DSPs in addition to the tester. These steps complicate the test development process.

Luckily, the transceiver itself is fully capable of performing these operations and they are typically implemented in hardware, so no additional software is needed. Thus, we propose to use the receiver of an identical transceiver to perform these complex steps on the load board.

1.4. ***Built-in EVM Measurement Set-Up***

As explained in the previous section, for EVM calculation, access to IQ signal information, rather than the decoded bit pattern is necessary. One solution that comes to mind would be to provide external access to points J and K, in the digital subsystem in Figure 2. However, this approach would require that 16 bits of data be taken off-chip. Moreover, these bits need to be interfaced with the tester, placing a bottleneck in terms of pin count. Our goal is to measure the EVM on-chip by accessing the output of the de-mapper (L and M) and provide a singe output that indicates the pass/fail state of the device.

1.4.1. *Proposed Built-in EVM Measurement Set-Up.* We propose to leave the job of synchronization, and similar computation intensive operations to the device designed to complement the DUT. The symbol pattern obtained at the output of the QAM mapper block (points L and M in Figure 2(a)) is used to estimate the EVM value.

Clearly, the receiver that is used to extract the data introduces some level of distortion. When the signal power is appreciable, as will be the case in a test environment, the EVM contribution of the receiver is typically small [Gu(2005)] and can be decoupled from the measurement result. Since the receiver on the load board can be fully characterized, the following simple expression is sufficient to eliminate the effect of the receiver.

$$EVM_{TX}^2 = EVM^2 - EVM_{RX}^2 \quad (2.1)$$

It should be noted that unless very accurate RF instrumentation is used to directly analyze the transmitter output, the devices on the load board always impose additional error, and these errors need to be decoupled from the measurement results in the same manner.

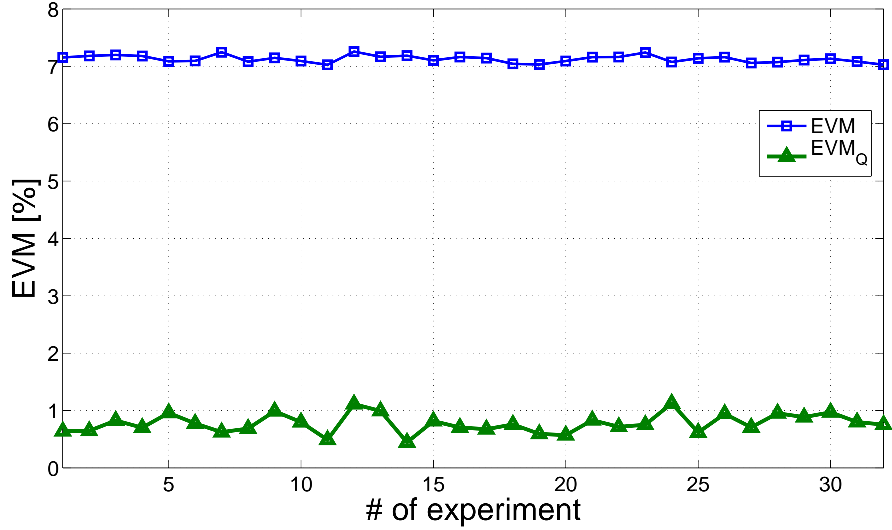


Fig. 6. EVM and quantized EVM comparison without DFT

1.4.2. *Built-in Measurement Concept.* Since we use the decoded symbol pattern (L and M) instead of using the signals at the output of the FFT block (J and K), the EVM value calculated using our method is slightly different from the standard EVM. We *virtually* map the decoded bit pattern to the constellation space and compare the locations of the received and transmitted symbols. If the received symbol is identical to the transmitted symbol, the instantaneous EVM is zero, and if they are not identical, the instantaneous EVM is given by the distance of these symbols. We call the resulting EVM value from such quantized information as EVM_Q.

At first glance, this calculation can be done without any changes in the digital subsystem at all. However, the measurement error in this case would be large. As a demonstration, Figure 6 shows the EVM measured through the traditional method by analyzing the IQ signals and EVM_Q measured by relying only on the decoded bit pattern but without any changes in the digital subsystem. Clearly, the error of EVM_Q measurement is too large

to be of practical use. This error basically is quantization error since symbols practically quantize the constellation space; larger symbol distances result in larger errors. One may consider reducing the overall signal power to reduce the symbol-to-symbol distance (eg. through an attenuator at the output of the transmitter). However, such an approach only sensitizes the device to noise, since error vector magnitude due to all other impairments (i.e. carrier leakage, IQ mismatch, non-linearity) are scaled with the signal power.

The key is to be able to reduce the symbol distances without reducing the signal power. To achieve this, we need to incorporate more symbols into the constellation. We call this process *constellation enhancement* and the ratio of the number of symbols in the enhanced constellation to the number of symbols in the original constellation as *constellation enhancement ratio* (CER). Luckily, this process requires only minor changes to the symbol mapper/demapper, which is a small component in the digital base-band [Cabral *et al.*(2006)], in addition to a few small digital blocks to enable the computation, as we will discuss in detail in the next section.

Figure 7 demonstrates the constellation enhancement concept. In Figure 7(a), we illustrate one symbol location in the original constellation together with the decision boundaries and some added noise around it. In Figure 7(b), the complex IQ plane is divided with finer granularity and the region corresponding to the original symbol location now yields 4 separate regions. Since decision boundaries are tightened and noise remained the same, some of the displaced symbols fall into the neighbor regions. As illustrated in Figure 7(b), displaced symbols are shared by the 2nd, 3rd and 4th regions in addition to its correct region. Symbols that fall in wrong regions are mapped into the middle of the corresponding region. For instance, portion of the shaded area that falls into the 3rd region are mapped to the center of the 3rd region. Similarly, parts of the shaded area falling into the 2nd and 4th regions

are placed at the center of the 2nd and 4th regions respectively. EVM of the symbols falling into the 1st region is zero, since all these symbols effectively fall on the correct constellation point. As such, the EVM associated with these symbols will be underestimated whereas EVM associated with symbols falling into the neighboring regions is overestimated. As the number of symbols increases, performance of the quantized EVM approaches to the standard EVM asymptotically. However, we do not need to improve the accuracy that much, since the standard EVM also has a certain amount of uncertainty. A more detailed explanation of the accuracy with respect to the number of constellation points is given in the next section.

1.5. *DFT*

The performance of the proposed Quantized EVM calculation depends on how finely the complex plane is divided, hence it depends on the number of bits per symbol. The maximum allowed number of bits per symbol in the normal mode of operation is 6. Therefore, the complex plane is divided into 64 regions. In the test mode, we increase this number by a factor given by the CER.

Another factor affecting the accuracy of our method is the level of EVM, which is a function of various types of impairments and noise. As the amount of impairment and noise increases, probability of a symbol falling in a wrong region increases and therefore contributes to the EVM calculated using our method. If impairment and noise is low, our method requires finer granularity in order to match the standard EVM measurement. The key is to make the right pass/fail decision based on EVM. Therefore, for circuits with an EVM value close to the standard limit and/or exceeding it, we would like to make accurate measurements. For circuits where the EVM value is much lower, underestimation does not

cause any misclassification. We make our design decisions on the DFT circuit keeping these constraints in mind. As an example, according to the IEEE 802.11 standard [iee(1999)], the limit on the relative constellation error for transmitter is -25dB, which corresponds to an EVM value of 5.6%. In the following section, we investigate the granularity level needed to match the results of our method to the standard method. In doing so, we try to match the measured EVM_Q value to the standard defined EVM value when the EVM is near 5.6%.

1.5.1. *Necessary DFT Modification.* The proposed method requires a slight modification to the symbol mapper/demapper block to support grid sizes that are smaller than defined in the communication standard, additional circuitry to calculate EVM on-chip, and a linear feedback shift register (LFSR) to generate the pseudo-random bit sequence.

In order to enable constellation enhancement, we need to increase the number of bits per symbol. To avoid any changes to the ADCs and DACs in the system, we still want to keep the 8-bit representation for the IQ signals. Thus, there is a limit to how far we can enhance the constellation. It should be noted that the limit for the overall resolution of the EVM_Q measurement is the same as the traditional measurement as long as the bit width limitation for I and Q signals is kept constant.

The constellation enhancement ratio is clearly proportional to the increase in the bit width of IQ symbol representation. A one-bit increase in the IQ symbol width yields a constellation enhancement ratio of 4. If we keep the 8-bit resolution for IQ signals, the maximum constellation enhancement ratio we can obtain is 64.

The overall DFT for built-in EVM computation is given in Figure 8. The required DFT modification to enable constellation enhancement is to change the symbol mapping/demapping blocks. We add another mode (test mode) into the symbol mapper/demapper table to support the enhanced constellation points. The coded IQ information is supplied

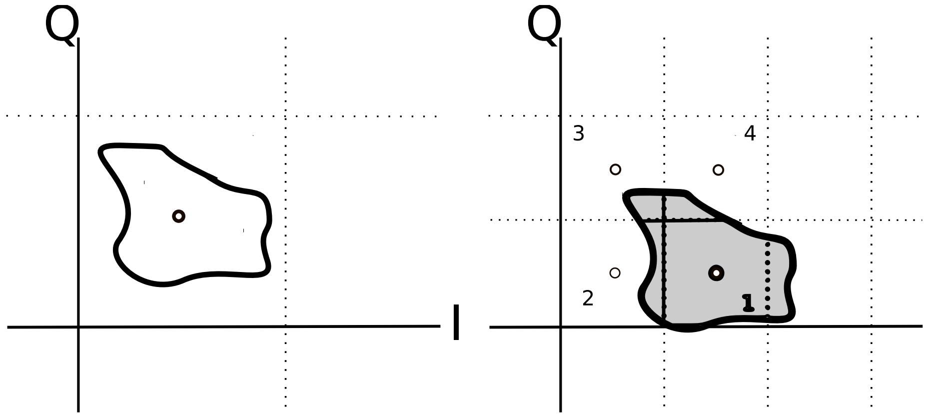


Fig. 7. Constellation enhancement by coding more symbols

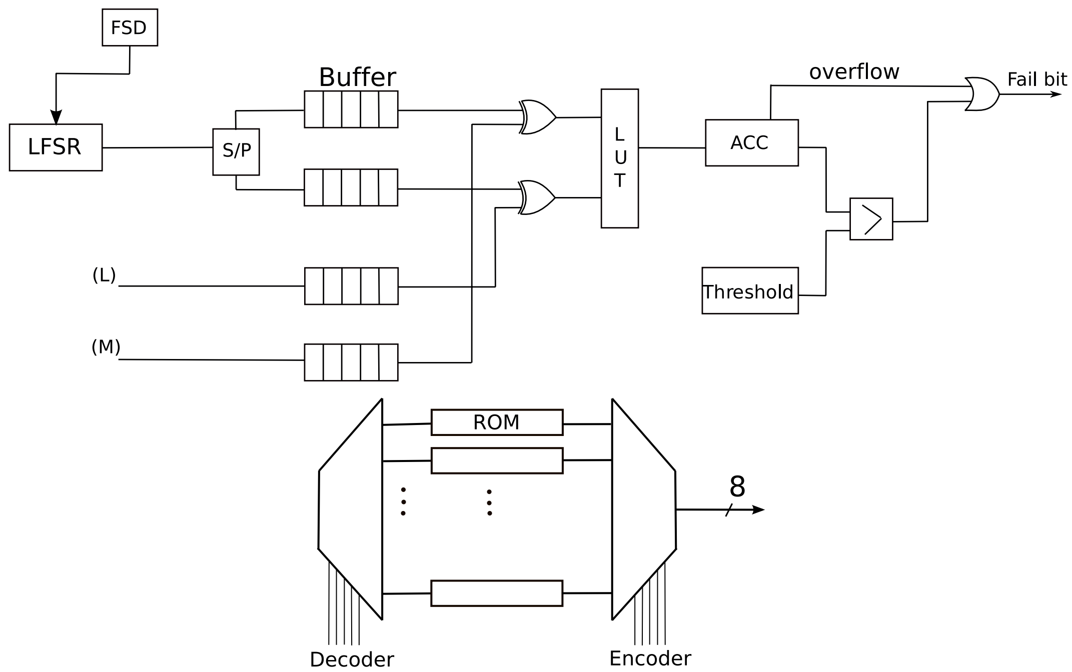


Fig. 8. Proposed DFT for built-in EVM measurement. (a) EVM computation block, (b) Symbol Mapper/Demapper

to the IFFT block in the same manner as in normal mode of operation and there are no additional modifications in the digital base-band.

In order to generate the same bit stream at the transmitter and receiver, we use an LFSR that is shared by the transmitter and the receiver. For the DUT, the LFSR generates the transmitter bit pattern, and for the golden receiver that performs the decoding operation, the LFSR provides the bit pattern to compare against. The generated bit pattern needs to be padded with long and short symbols to enable the receiver synchronization. These are deterministic symbols, and are stored in a small piece of memory to add to the incoming bit pattern.

In order to enable the EVM computation, additional digital blocks are necessary, as shown in Figure 8. We propose an innovative method to calculate EVM using a small set of digital circuitry. We exploit the coding scheme employed to transmit the signals to calculate EVM in a very efficient way. Gray coded symbols differ only one bit from each neighbor on the constellation diagram. Hence, dislocation of a symbol to a neighbor region results in a single bit flip in the bit representation of the symbol. Moreover, the coding can be adjusted in a way such that second and third neighbors can be detected by counting the number of bit flips in the decoded IQ symbols. Using this property, we can calculate the amount of displacement in terms of constellation regions by simply checking the number of bit-flips in the received symbol. Since we obtain I and Q channel components of each received symbol (M and L points), we can estimate the amount of displacement in each direction. For instance, if the number of flips in the Q channel of a symbol is one, received symbol is displaced by one symbol in Q direction. Note that we do not need the direction of the displacement; hence, using only the number of bit flips is adequate to obtain the displacement. Similarly, bit flips in I channel component of a symbol is due to displacement in I direction in the IQ plane. The amount of displacement is obtained through determining the number of bit flips in both of the received channels.

The EVM calculation process is activated by switching the transceiver in testing mode. As explained previously, the transmitter uses a pseudo-random test sequence as input in the test mode. The same pseudo-random sequence is generated by using the same configuration of LFSR in the receiver. The generated test sequence is fed directly to XOR blocks. The other input of the XOR gates comes from de-mapper block of the receiver. The output of the the XOR block is sent to a look-up table (LUT). We use an LUT to map the displacement in terms of flips to the actual amount of displacement in the IQ diagram. According to the constellation model we chose, each constellation point is separated by 1 unit in horizontal and vertical direction, while nearest diagonals has an error energy of 2. The LUT is filled with the Euclidean distances of the constellation points. The input to the LUT is the bit errors, while the output is the square of the error vector magnitude of the received symbol. The output of the LUT is fed to an accumulator. Therefore, the square of the error vector magnitude is accumulated to give the total error in the received signal. This accumulated value is compared to a predetermined threshold level in order to decide if the circuit passes or fails the EVM test. If the accumulator overflows due to excessive EVM, a fail signal is sent irregardless of the comparator output.

One way of determining the threshold value is to use the energy of the pseudo-random sequence generated by LFSR. Since the configuration of the LFSR circuit is known by the designer, the generated test sequence and its energy are also known. Hence, using the allowed amount of EVM degradation, it is possible to calculate the threshold level. Alternatively, one can use the average power of the constellation. Equation 2.2 relates the average power of the constellation for the nearest horizontal/vertical neighbor distance of 1 unit to the number of symbols, S, in the constellation. Imposing the allowed limit on EVM degradation on the calculated average constellation power yields the threshold level.

In order to keep the size of the accumulator at a minimum level and to prevent the capacity of accumulator to affect the result, we use the overflow bit of the accumulator to set the fail bit.

$$P_{av} = \frac{S-1}{6} \tag{2.2}$$

1.6. *Analytical Model*

Since the proposed DFT needs to be incorporated into the design flow, we need to determine the necessary constellation enhancement ratio (CER) before the analog/RF subsystem is finalized. In order to enable this up-front DFT, we develop an analytical model based on the specified EVM limit for the transmitter.

The worst case measurement error for EVM_Q happens when all the EVM is due to noise only and there is no displacement for the symbols due to other impairments. While we will later show some simulation results to this point, intuitively this phenomenon can be explained as follows: impairments such as carrier leakage, filter characteristics, and phase offset displace the symbols from their original locations while noise generates a band around it. Due to the original displacement, the probability that the enhanced symbols cross the decision boundaries becomes higher.

Thus, our formulation juxtaposes the EVM limit on a Gaussian noise model and this model is used to calculate and compare the EVM_Q value with the EVM value to determine a pessimistic CER.

This formulation is developed for no-translation and no-rotation without loosing the generality; these terms can be easily incorporated by transforming I_i and Q_j in the equation. Figure 9 illustrates derivation of the analytical model. Suppose that a symbol with

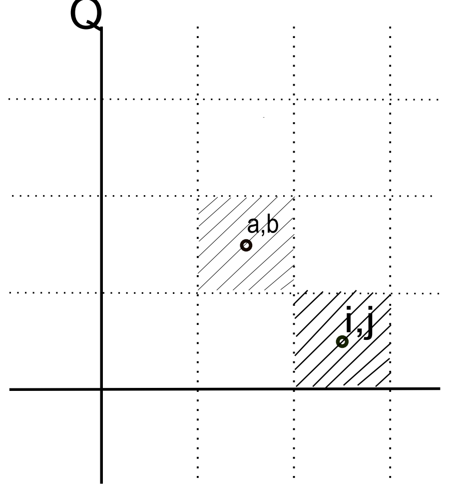


Fig. 9. Analytic quantized EVM estimation

an ideal location of (I_i, Q_j) is sent and we would like to estimate the probability of the received symbol falling on the constellation point (I_a, Q_b). We can estimate that probability by integrating a Gaussian function centered at (I_i, Q_j) with noise N_0 inside the acceptable region of the point (I_a, Q_a). Calculating this probability for all constellation points, weighing them with the square of the distance from (I_i, Q_j) and averaging them with the total number of points yields the mean square error for the constellation point (I_i, Q_j). Finally, using outer loops of equation 2.3, we calculate the overall mean square error of the system. Equation 2.3 is derived for square shaped constellation for the sake of simplicity, without loosing the generality. Where, M represents the size of the constellation, $I_{a,min}$ and $I_{a,max}$ represent the boundary regions of ath point in I axis, $Q_{b,min}$ and $Q_{b,max}$ represents the boundary regions of bth point in Q axis, N_0 represents noise level and EVM_0 represents the total power of the constellation.

$$EVM'_Q = \sum_{i=1}^{\sqrt{S}} \sum_{j=1}^{\sqrt{S}} \sum_{a=1}^{\sqrt{S}} \sum_{b=1}^{\sqrt{S}} ((I_i - I_a)^2 + (Q_i - Q_b)^2) \int_{I_{a,min}}^{I_{a,max}} \int_{Q_{b,min}}^{Q_{b,max}} N_2(I_i, Q_i, I_a, Q_b, N_0) dxdy \tag{2.3}$$

$$EVM_Q = \frac{EVM'_Q}{EVM_0} \tag{2.4}$$

We use equation 2.3 to calculate the minimum constellation enhancement ratio that provides the desired accuracy. Equation 2.3 approximates the EVM calculation method as the number of constellation approach to infinity. However, we do not have to approximate the standard EVM value very accurately, since it has an inherent uncertainty due to the randomly applied input. We need to measure EVM_Q within the band of uncertainty of EVM measurements in general.

We take the worst case for CER calculation, where there is only Gaussian noise and no impairment in the system. Impairments move constellation points close to decision boundaries and result in a higher symbol error rate, therefore better performance for our method. Hence presence of impairments yields a lower CER.

To demonstrate that this analytical model correctly determines the necessary constellation enhancement ratio, we compare the results from the analytical model to the actual value of EVM. Figure 10 shows one example when all of the EVM is caused by noise in the system (worst case), as we have taken in the analytical model. In this example, the actual EVM is 10%. Based on the analytical model, we see that a constellation enhancement ratio of 16 is needed to estimate the EVM within 1% error (within the bounds of uncertainty of EVM). Simulations also confirm this result. In the next section, we will show that when

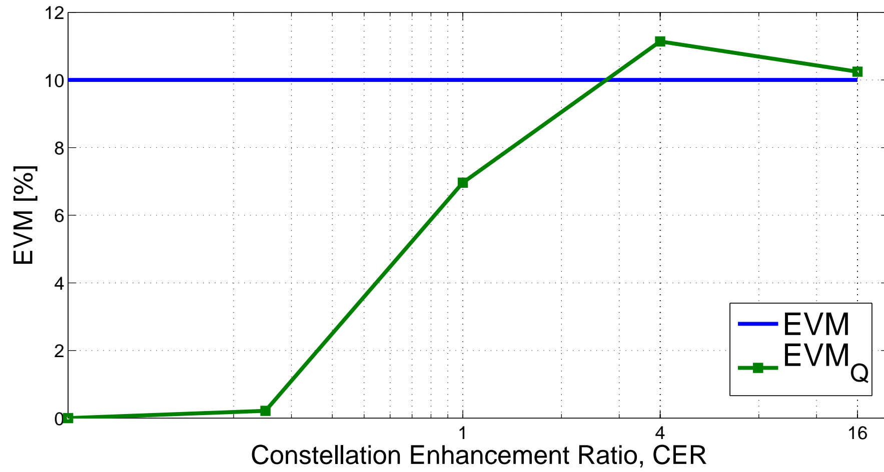


Fig. 10. EVM_Q and EVM vs CER

there are impairments in the circuit, the necessary constellation enhancement ratio is less than what we determined up front. However, we do not believe that it would be good engineering practice to rely on the *badness* of the circuit.

The interesting behavior of the accuracy of EVM_Q measurement at CER of 4 can be explained as follows: as the CER increases, the measured EVM value increases. At the crossover point, the IQ plane is divided finely enough that only a small number of symbols cross over, but their effect on EVM is large since once the symbols cross over, they are mapped onto the next symbol location. When the CER is increased further, the overestimation reduces and the EVM_Q accuracy increases monotonically. Based on this analytical model, we determine that we need a CER of 16 to effectively measured the EVM for WLAN systems (max EVM of 5.6%).

Note that the analytical model developed in this section does not take the effect of gray code based distance estimation method. In this model, we assume that we know the displacement amount in terms of minimum symbol separation. However, gray code based distance calculation method does not yield the exact displacement amount. In the experimental result section, we show that this effect is negligible for the selected CER. In the next section, we provide the overhead analysis for a CER of 16.

1.7. *DFT Overhead*

Our proposed DFT technique requires changes to only very small blocks in the digital subsystem. Nevertheless, we provide a detailed overhead analysis in this section.

In the testing mode, a larger constellation is used for the reasons mentioned in the previous sections. For that purpose, mapper and de-mapper needs to be modified to support a larger constellation size. This enhancement is achieved by employing one 5-to-32 decoder for each I and Q channel and connecting the outputs of the decoder to the select input of 32 ROMs that contain 8 bit gray codes. Hence, I and Q channels are gray coded using the gray codes in the ROMs. The code read from the RAMs are transferred to the FFT block

via an encoder. Schematic of gray encoder for test mode is shown in Figure 8(b). Overhead of the gray coding circuitry is given in Table 1. Another block we use in the transmitter is a 10-bits LFSR. Contribution of both LFSR blocks that are used in receiver and transmitter is shown in Table 1.

On the receiver side, we use an S/P and four 5 bit buffers, as well as 2 5-bit XOR blocks. Contribution of these components are given as the *other* category in Table 1. An 11-bit accumulator is employed to handle the error magnitude of 320 test symbols, contribution of this component is not significant. Table block is a 5x5 LUT, that contains 6 bits in each entry to represent the error value. Note that the maximum amount of detectable distance is 5, so the largest amount of detectable error is 50 according to our error calculation scheme. Therefore 6 bits in each entry is adequate to represent the error. The output of the accumulator is compared to the threshold level of the allowed EVM and fail bit is set if accumulated error is larger than the threshold level or the overflow bit of the accumulator is set.

1.8. *Experimental Results*

Increasing the number of constellation points has shown to be effective in improving the performance of our proposed method in previous sections. In this section, we provide simulation and hardware measurement result to validate the viability of our method.

1.8.1. *Simulation Results.* The simulation setup employed to verify the proposed method is shown in Figure 11. This figure includes a typical WLAN OFDM system. Digital subsystem encodes the input bit stream using OFDM coding scheme and the output signal is converted analog domain through 8 bit digital analog converters (DAC). OFDM modulated analog signals are then up-converted to frequency ω_0. Gain impairment, phase

	w/o DFT [mm^2]	w DFT [mm^2]
Digital & Symbol	1.4 [Perels *et al.*(2005)]	1.4
RF	17.2 [Maeda *et al.*(2006)]	17.2
Transmitter/Mapper	0	0.00328
Receiver/Mapper	0	0.00328
Table	0	0.0003
LFSR	0	0.002
Others	0	0.00016
Total	18.6	18.607
Change	0 %	0.0485 %

Table 1. DFT overhead

impairment, carrier leakage and local oscillator phase noise are also injected to generate impairments for the transmitter. Before the RF signal is fed back to the receiver, we super impose additive white noise (AWGN) on the signal. Power of the noise is selected so that BER is 10^{-3} at the out of the receiver. A similar set-up is used for the receiver. Standard EVM is calculated using the data obtained at the output of the receiver FFT block. While quantized EVM is calculated using our proposed method.

Figure 12 compares the EVM result calculated using the standard method and the proposed method. Lines represent the mean of the calculated values, while error indicators shows the 99.5% (3σ) confidence intervals of the curves. As shown in [Acar *et al.*(2008)], and confirmed through our experiments here, the standard EVM is not a value that can be calculated with 100% accuracy. There is an uncertainty in EVM due to randomly generated

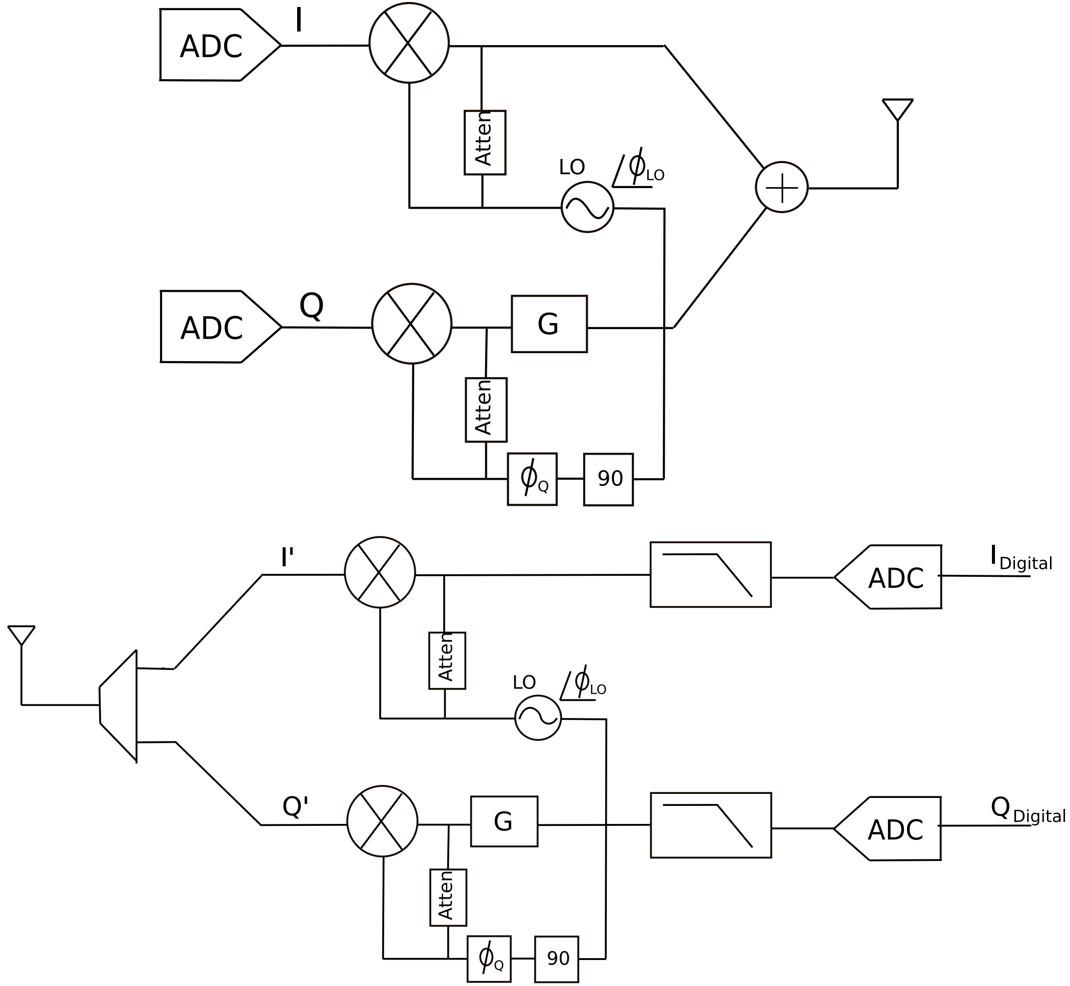


Fig. 11. Simulation Set-up (a) Transmitter, (b) Receiver

data. The EVM curve generated using our method works poorly for small constellations; However, after a certain point it matches the standard EVM curve and remains in the uncertainty band for increased number constellation points. Hence, there is an optimum number of constellation points that minimizes the DFT cost while maintaining the required level of accuracy. In Figure 12(a), results of two methods are compared with the influence of AWGN noise only. Figure 12(b) confirms the result from the analytical model that, a CER of 16 is adequate to measure EVM accurately enough in the absence of impairments.

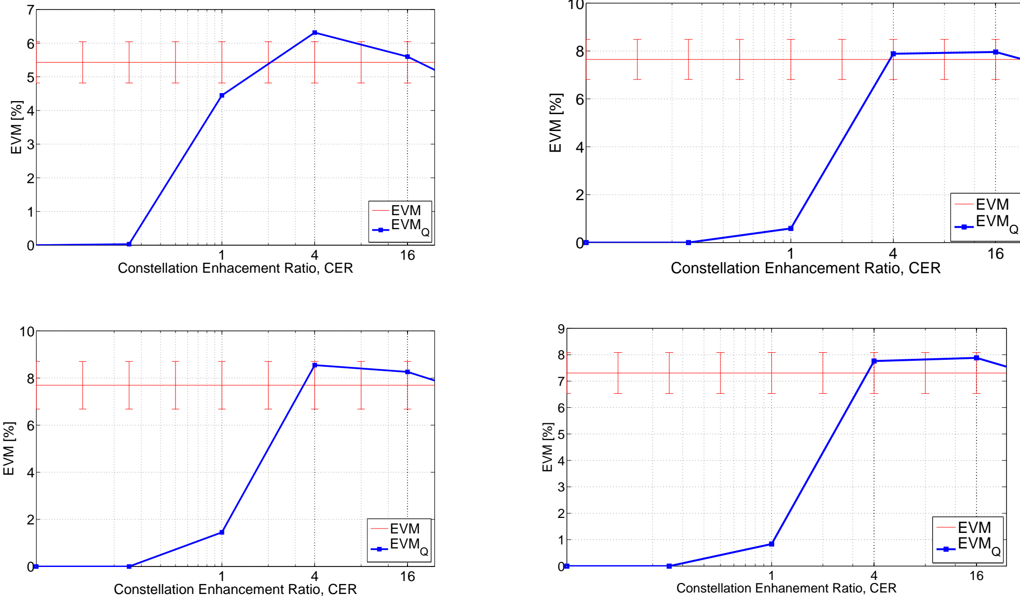


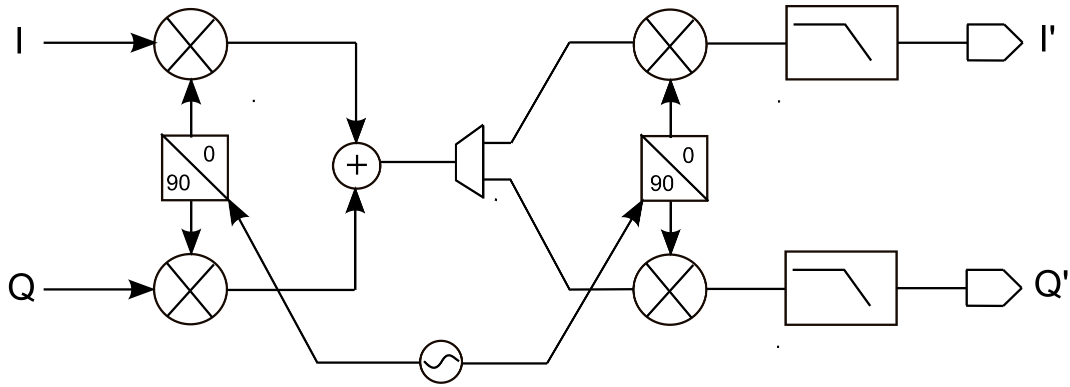
Fig. 12. Simulation results for various types of impairments. (a) Only noise, (b) 8% gain impairment, (c) 5 degrees of phase impairment, (d) 5% gain and 3 degrees of phase impairment.

Introducing impairments exacerbates EVM values, however, constellation size needed to make our method accurate enough reduces. This is expected since impairments result in larger mislocations and therefore increases the probability of a symbol falling in the neighbor constellation region. Since our method relies on symbols falling in wrong regions, performance of our method reaches the standard method for smaller constellation sizes. Figures 12(b) and 12(c) show the EVM comparison for 8% gain and 5% phase impairment respectively. We also incorporated local oscillator phase noise and carrier leakage effect for these figures. In Figure 12(d), we show the combined effect of the gain and phase impairments as well as LO phase noise and carrier leakage. Gain and phase impairments are selected to be 5% and 3 degrees respectively. A typical value of -70 dBc/Hz is used for

phase noise level and -40dB isolation level is selected between RF and LO signal. These simulation results also confirm that the necessary CER is 16 to measure the EVM within an error of 1%.

1.8.2. *Hardware Measurement Results.* We have used off-the shelf components for the hardware measurement set-up. Since the degradation in the modulation quality is mainly due to the analog part of the transmitter, we constructed the IQ modulation part only using hardware components. The schematic of the transmitter system built for hardware measurement is shown in Figure 13(a). The base-band input waveform is generated using Matlab and fed into hardware set-up using an arbitrary wave generator (AWG). The output of the receiver is measured and digitized by an oscilloscope and converted back to digital domain. Digital data are processed in Matlab to obtain the original data back and the measure the EVM. The measurement set-up of this step is exactly the same as the set-up used in the previous section, except that IQ modulation/demodulation is performed using hardware components. In all these steps, we have kept the 8-bit resolution limit for the IQ signals. Figure 13(b) shows the picture of the implemented IQ modulator.

The results obtained from the hardware measurements are shown in Figure 14. The straight line shows the mean of the standard EVM measurement and superimposed error bars on this line show its 99.5% (3σ) confidence interval. The EVM_Q curve is obtained using our proposed method. These results show that EVM values that are calculated using our method match the standard EVM measurement at a CER value of 4 and remain in the acceptable accuracy level for the increasing number of constellations consistently. The reason that this hardware set-up requires a lower CER for the same accuracy is that its EVM is much larger compared to the standard defined EVM.



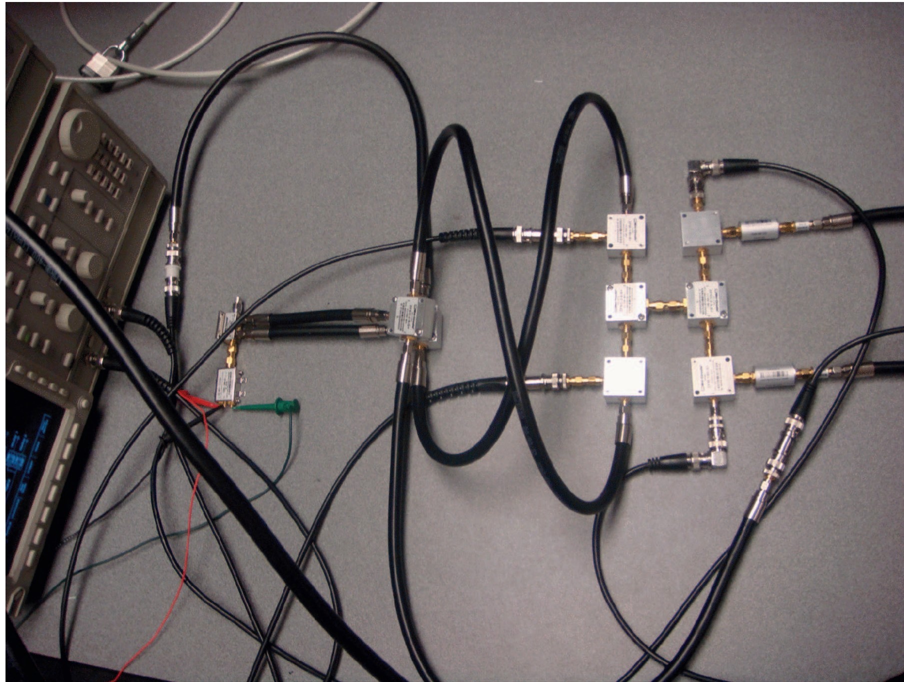


Fig. 13. Hardware Set-up. (a) Schematic, (b) Implemented transmitter

1.9. *Summary*

In this section, we presented the constellation enhancement technique to enable low-overhead built-in measurement of EVM for OFDM transmitters. EVM measurement requires complex operations that need to be implemented at the tester, placing a burden on its computational resources.

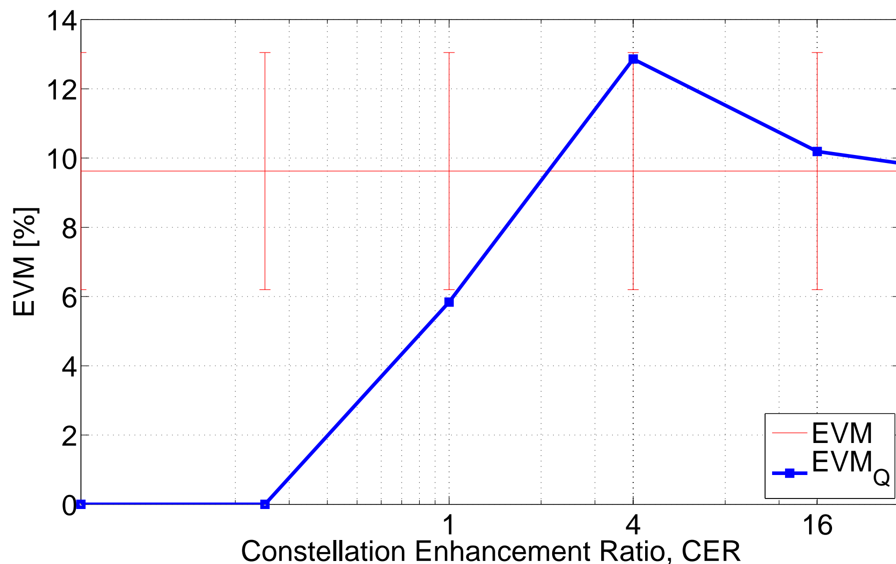


Fig. 14. Hardware measurement results

We observe that the transceiver itself is designed to perform fully complementary operations in the base-band, and it includes hardware to perform these complex operations that otherwise would generate a computational bottleneck. As such, we propose to use the receiver of an identical device on the load board to enable a low-cost EVM measurement. Our EVM measurement method relies on the decoded bit pattern. One problem with using the decoded bit pattern arises from the fact that the receiver fully decodes the bit pattern, and the resulting information is typically of little use to measure EVM with an acceptable accuracy. We solve this problem by incorporating a small DFT change in the digital baseband to code more symbols into the constellation, thereby increasing the resolution and reducing the error in EVM measurements. We propose an innovative method of measuring EVM on-chip with very little overhead by exploiting the symbol coding scheme.

We show through experimentations conducted using a MATLAB model and hardware measurements conducted using off-the-shelf transceiver components, that the accuracy of our technique is within the uncertainty band of EVM with very reasonable area overhead.

2. Accelerated Statistical Pre-silicon Evaluation/Analysis

Defective Parts Per Million (DPPM) is an important quality metric that indicates the ratio of defective devices shipped to the customers. It is necessary to estimate and minimize DPPM in order to meet the desired level of quality. However, DPPM estimation requires statistical simulations, which are computationally costly if traditional methods are used. In this work, we propose an efficient DPPM estimation method for analog circuits that greatly reduces the computational burden. We employ a model based approach to selectively simulate only consequential samples in DPPM estimation. We include methods to mitigate the effect of model imperfection and robust model fitting to guarantee a consistent and efficient estimation. Experimental results show that the proposed method achieves 10x to 25x reduction in the number of simulations for an RF receiver front-end circuit.

2.1. *Background*

Due to process variations, performance parameters of manufactured devices are probabilistic and can be represented with a joint probability distribution function $JPDF(s)$, where s is the vector of specification parameters. DPPM can be defined as the integral of the JPDF using equation (2.5).

$$DPPM = 10^6 \int_{s \in A_{DPPM}} JPDF(s) ds \tag{2.5}$$

where A_{DPPM} is the region of specification space where instances resulting in DPPM reside. This equation cannot be evaluated deterministically due to the high dimensionality of the space and the fact that the exact form of the JPDF is not known.

The most widely used method for computing statistical parameters such as DPPM is MC simulations due to its ease of implementation. Monte Carlo formulation of DPPM is given in (2.6).

$$DPPM_{MC} = \frac{1}{N}\sum_{i} 1(X_i is\ DE),\ X_i \sim JPDF(p) \tag{2.6}$$

where X_i is sampled from $JPDF(s)$, N is the total number of simulations, and $1(DE)$ is the indicator function that returns 1 if its associated condition, DE (Defective Escape), is true and 0 otherwise. Monte Carlo method yields an unbiased estimate of DPPM and the variance can be estimated with equation (2.7), which is inversely proportional to $\sqrt{N}$, where N is the number of simulations.

$$var(DPPM_{MC}) = 10^6 \frac{DPPM_{MC}}{N} \tag{2.7}$$

Monte Carlo method enables to estimate DPPM in high dimensional space, but it is inefficient, because sampled region is typically wide and most of the samples do not fall in the probable region of the defect escapes. For example, if the defect escape level is on the order of $10^{-3}(DPPM = 1000)$, only 1 out of the 1000 simulations contribute to DPPM in MC simulation. Therefore most of the effort is spent in simulating devices that bear no useful information. This is not surprising since MC is a general purpose method and can be used to estimate any statistical parameter at a cost of large number of simulations.

Importance sampling is a generic name for a class of methods that aim at only generating instances that are relevant in order to improve efficiency. One way of achieving this efficiency boost is to sample from an alternative distribution such that the probability of sampling defect escapes is increased. This can be explained trough an example illustrated in Figure 15. **p** is the original distribution that device instances are sampled, and p^* is the probability

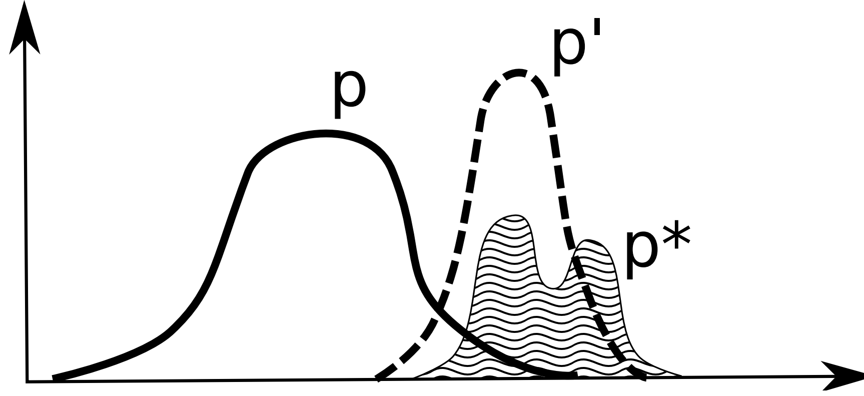


Fig. 15. Estimation variance can be reduced by using a more appropriate sampling distribution, such as p' instead of p. The optimal sampling distribution is p*.

distribution of the defect escapes. The figure shows that only a small fraction of the generated samples overlap with the defect escape region. Hence, most of the samples do not contribute to the estimated parameter in this example. Efficiency can be improved by sampling from an alternative distribution that would increase the chance of generating defect escape instances, such as from distribution p'. This is due to the fact that the overlapping region of p' and p* is larger. Using a different distribution results in bias, but it can be easily removed through properly weighing the generated instances. Importance sampling formulation of DPPM is given in (2.8).

$$DPPM_{imp} = 10^6 \int_{\forall P} 1(DE) \left[\frac{JPDF(p)}{JPDF'(p)}\right] JPDF'(p)dp \tag{2.8}$$

where $JPDF'(p)$ is the alternative distribution. Equation (2.8) can be interpreted as the weighted integral over the alternative distribution, as shown in (2.9).

$$DPPM_{imp} = 10^6 \int_{\forall P} 1(DE)\, W(p)\, JPDF'(p) dp \tag{2.9}$$

$$W(p) = \frac{JPDF(p)}{JPDF'(p)} \tag{2.10}$$

W(p) is the weighing function that removes the bias. Finally, we put the equation above into discreet form.

$$DPPM_{imp} = \frac{10^6}{N} \sum_i 1(DE) W(X_i),\ X_i \sim JPDF'(p) \tag{2.11}$$

Importance sampling estimate is unbiased and its variance is given in (2.12).

$$var(DPPM_{imp}) = \frac{1}{N} E\{1^2(DE)W(p) - DPPM^2_{imp}\} \tag{2.12}$$

It has been shown that importance sampling method can reduce and even eliminate variation if an appropriate alternative distribution is found [Gentle(2003),Srinivasan(2002)]. This can be shown on the example illustrated in Figure 15. The optimal sampling distribution in this example that would yield minimum variance is p*, since all sampled instances fall in the desired region.

In the digital domain, importance sampling is used to determine the fail probability of various structures, such as SRAMs [Kanj *et al.*(2006)]. Importance regions are determined based on the relation between the circuit speed and process parameters. For instance, it is known that circuits get slower when the gate length is increased. Thus, in order to estimate the fail probability based on speed, the gate length distribution should be shifted up. Unfortunately, there is no systemic way of determining the importance sampling region. This problem is especially though when multiple specifications need to be evaluated at the

same time. Aside from the complexity of relations between the specifications and the process parameters, they may have conflicting requirements for importance regions.

2.2. *Methodology*

Finding the alternate distributions to generate important samples is a challenging task in high dimensions. We propose an innovative method to sample from the desired region to improve efficiency without finding an optimal distribution. We combine the MC simulator with a model based behavior estimator to form an importance sampler. This method filters out samples that are not relevant for the statistical parameter that we wish to estimate This effectively enables us to sample from the desired region without knowing its distribution.

In order to achieve this goal, we first generate a training set of sample devices to form a model predictor. This predictor need not be extremely accurate since we will use guard-banding to reduce the impact of inaccuracies. However, it is important that it has a very small number of outliers. In other words, there should be very few evaluation points with gross prediction errors, whereas small prediction errors are inconsequential. Once the model is formed, we generate a large number of MC samples and use the model to predict which ones of these samples are potential contributors to DPPM. Only these potential contributors are simulated to estimate the overall DPPM accurately.

A conceptually similar approach of using a model for sample selection has been employed as a part in a tail probability estimation method [Singhee and Rutenbar(2008),Stratigopoulos and Mir(2010)]. However, this method is devised for single dimensional tail distribution fitting, and cannot be used for DPPM estimation. A simple classifier as a statistical blockade suffices in [Singhee and Rutenbar(2008), Stratigopoulos and Mir(2010)] since missing samples that fall into the tail during the fitting phase is not catastrophic as long as these

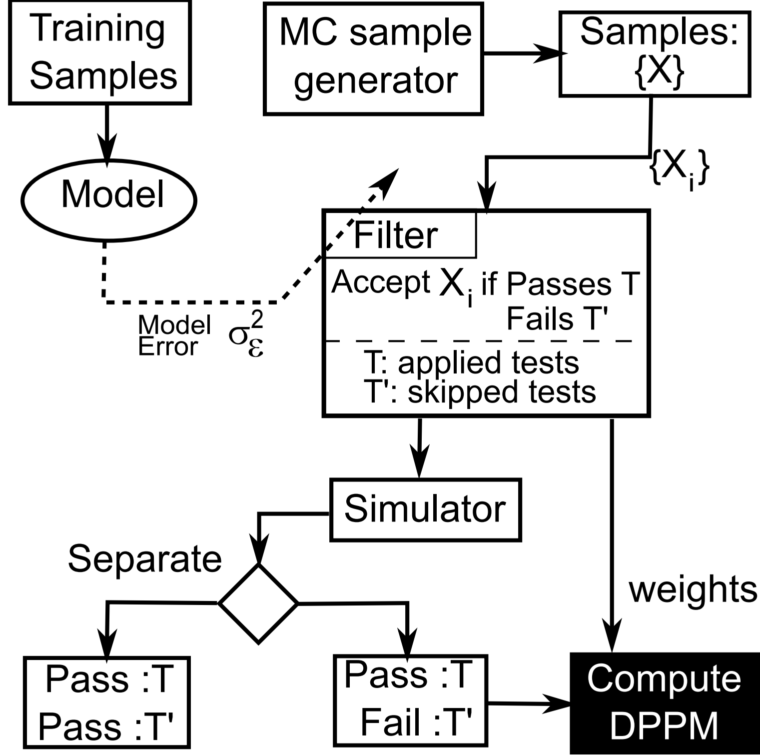


Fig. 16. Model Based Filtering Flow

misses do not bias the samples in the tail. For our purpose of model-based filtering, small model errors are acceptable. Whereas, gross modeling errors result in misprediction.

Note that we do not rely solely on the model response and we do not assume that the model is working perfectly. Otherwise there would be no reason to use the simulator. Instead, we take precautions by defining guard regions to eliminate wrong decisions. The flow of the proposed method is given in Figure 16.

Since we only simulate potential DPPM contributors, we need to assign weights to filtered samples to remove the bias. We explain the weight selection method through the algorithm we use to filter important samples, shown in Algorithm 17. These weights are also used to estimate the variance of our technique. As explained earlier, variance is a measure of robustness.

```
k=0
do {
p=next MC sample
k++
} while [(s=Model(p)) isNot defectEscape]
```

$p_{imp,j} = p$

$W_{imp,j} = \frac{1}{k}$

Fig. 17. Filtering and weight assigning algorithm

The variable "k", indicates the number of MC samples examined until a potential DPPM contributor is found and simulated. The first sample in the MC set is selected and its behavior is estimated using the model. The algorithm moves to the next MC sample if estimated behavior does not potentially result in a defect escape and repeat this procedure until a sample with desired properties is found. Then, the instance is added to the important device set and it is assigned $\frac{1}{k}$ as weight. The algorithm is used repetitively to shrink the large set MC samples into a much smaller set.

Once the important sample set is obtained, it is simulated and DPPM value is estimated according to the simulation results using the equation below.

$$DPPM = \frac{10^6}{M} \sum_{j}^{M} W_{imp,j} 1(DE) \tag{2.13}$$

2.3. *Model Generation*

Sampling from the desired region of the input space relies on the generated model that maps process level input parameters into performance parameters. This model can be generated in many ways, such as Response Surface Modeling, Artificial Neural Networks

(ANN), and Support Vector Regression. In this paper, we chose to use ANN method due to its ease of implementation. We used neural network toolbox of Matlab [mat(2010)] to fit a regression model between process level parameters and performance parameters. The most popular network topology for function estimation is 2 layer feed forward network, where typically sigmoid and linear function are used as activation functions. However, we have observed that this architecture does not yield the best results due to multidimensional and nonlinear nature of the problem. Hence, we employed a 3 layer topology and used radial basis, sigmoid, and linear function in the hidden layers respectively.

The network is trained by using a sample set of representative device instances. We generate a sample set of devices in process space (p_{train}) using MC sampling method and simulate them to get their corresponding response (s_{train}). Then, the network is trained to achieve a good fit between the input (p_{train}) and output parameters (s_{train}) by assigning appropriate weights. We used one of the most widely used methods in feed forward topology, the gradient descend method, to assign the network weights. Training, testing and verification sets are assigned to have 90%, 5%, and 5% of the total training sample population, where samples are randomly assigned to these groups. Three layer approach successfully models high dimensional system even for nonlinear performance parameters. In order to ensure a good fit, we use multiple training sessions and use the network that provides the best fit.

2.4. *Adaptive Guard-banding and Compensating for Model Error*

Although the generated ANN model produces fairly accurate estimates, error in estimation is inevitable. A compensation mechanism to suppress the impact of model estimation error is necessary to prevent bias in DPPM estimation. We will achieve this goal through

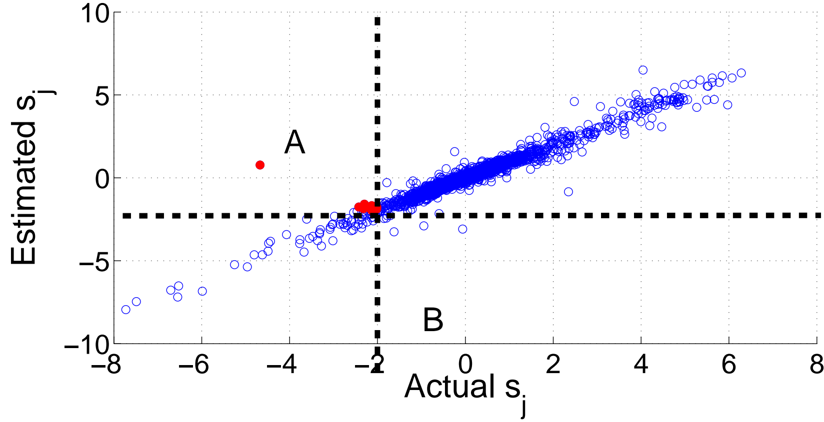


Fig. 18. Model error results in mispredicted circuit behavior. Mispredicted samples are shown in region A and B

adaptive guard-banding. First, let us explain the need for guard-banding. Suppose that for simplicity, our goal is to estimate the fail probability of one specification using our technique. Figure 18 shows the fitting plot between the estimated parameter and the actual parameter. The specification limits are given in dotted lines. Out of the four quadrants shown, two quadrants contribute to error, which are shown by regions A and B. B is the region where good devices are predicted as bad due to the modeling error. This type of error does not have an impact on DPPM estimation, since samples that are predicted as fails are simulated. As a result, at the end of the simulation, we will find the actual outcome of these samples. Therefore, the only effect is increasing the number of simulations. However, region A type errors are influential. Since devices falling in this region are predicted as good circuit and not simulated, they contribute to error in computing the fail probability. Therefore, the main concern in model error compensation is to reduce the number of devices falling in region A.

In order to avoid region A type errors, we introduce guard bands to reduce the probability of a wrong decision. In order to identify a sample as a defect escape candidate, its model response has to pass the tests that are in the measurement list and fail any of the tests that are not measured. Decision process is illustrated in Figure 19, where Figure 19(a) shows the distribution of a specification that is included in compacted list and Figure 19(b) shows a distribution of a specification that is not measured. Dashed vertical lines show the acceptable limit of the specifications. Important samples that potentially contribute to DPPM ideally pass according to the measured specifications and fail at least one of the unmeasured specifications. However, some of the instances may be misclassified if the model is not perfect. Influential ones are that fall close to the specification limits, since they can easily move to the wrong region if the model makes an error.

It is important to note that guard bands are imposed in opposing directions for specifications that are included in the test list and specifications that are not included in the test list. Since our goal is to reduce the chances of missing a sample that passes the tests in the list but fails the unmeasured specifications. Guard bands for measured specifications extend the specification limit while guard bands for unmeasured specification tighten the specification limits.

Guard-banding invariably will increase the number of inconsequential samples that need to be simulated. As a result, it will decrease the efficiency of the proposed technique. In this work, we use an adaptive guard banding approach. The amount of guard band is determined based on the modeling error, which we can estimate from the training set. We use the RMS modeling error for samples that fall near the specification limits to impose the guard bands. Guard band limits are illustrated with dotted vertical lines in Figure

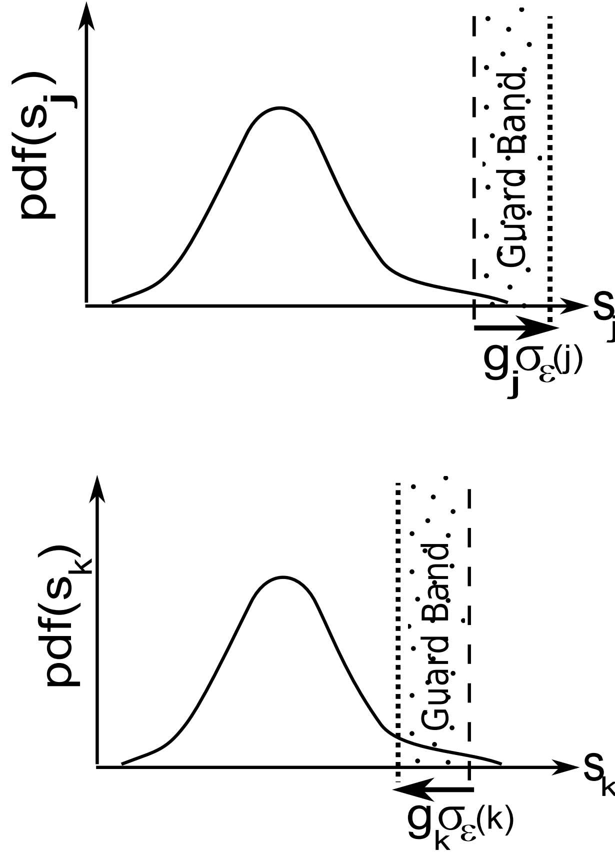


Fig. 19. Introducing guard band prevents the model from making wrong decisions.

19, where g_i is the guard band coefficient setting the extent of the band and $\sigma_\epsilon(i)$ is the standard deviation of the residual error, defined in equation (2.14), for the i^{th}specification.

$$\sigma_\epsilon^2 = \frac{1}{N_T} \sum_i (\hat{s}_i - s_i)^2 \tag{2.14}$$

2.5. *Robust Model Generation*

The neural network model predicts performance parameters of the training samples with a high success rate. However, since the training set size is typically small, the generated model may not be very successful in generalization, since the space that the training set

covers may not be adequate to generate a robust fit. For instance, if a sample resides at a point that is not covered by the training set samples in the input space, the model may fail to correctly predict the result. Figure 20(a) shows the response of the ANN, whose fitting parameter was 99.9% for the training set, but has greatly degraded to 90% when applied on a large disjoint set of verification samples. The figure indicates that the ANN has a poor performance in generalization.

We improve the robustness of the generated model by training with samples that are generated from a wider input space. Process level parameters are typically assumed to be Gaussian, hence we sample from artificially widened Gaussian distribution. This enables us to cover a wider space and therefore robust estimation. Figure 20(b) shows the response of the ANN that is trained with widely sampled set. The prediction performance is greatly improved on the same large disjoint verification set.

2.6. *Results*

We apply the proposed method to an RF receiver front-end circuit, which consists of a source degenerated cascode LNA (Low Noise Amplifier) and a double balanced Gilbert mixer. The schematic of the circuit is given in Figure 21. Process variation is injected in length and threshold voltage of the active devices and to passive components. Injected process variation is 10% for active devices and 15% for passive devices. Specification parameters of the circuit are input and output matching ($s_{11}$and s_{22}), gain (s_{21}), bandwidth, center frequency, noise figure, power consumption, 1dB compression point, $3^{rd}$and 2nd or order input referred intercept point.

Performance of the proposed method is evaluated and compared with the Monte Carlo method and a re-sampling based method [Stratigopoulos *et al.*(2009b)]. 100k device in-

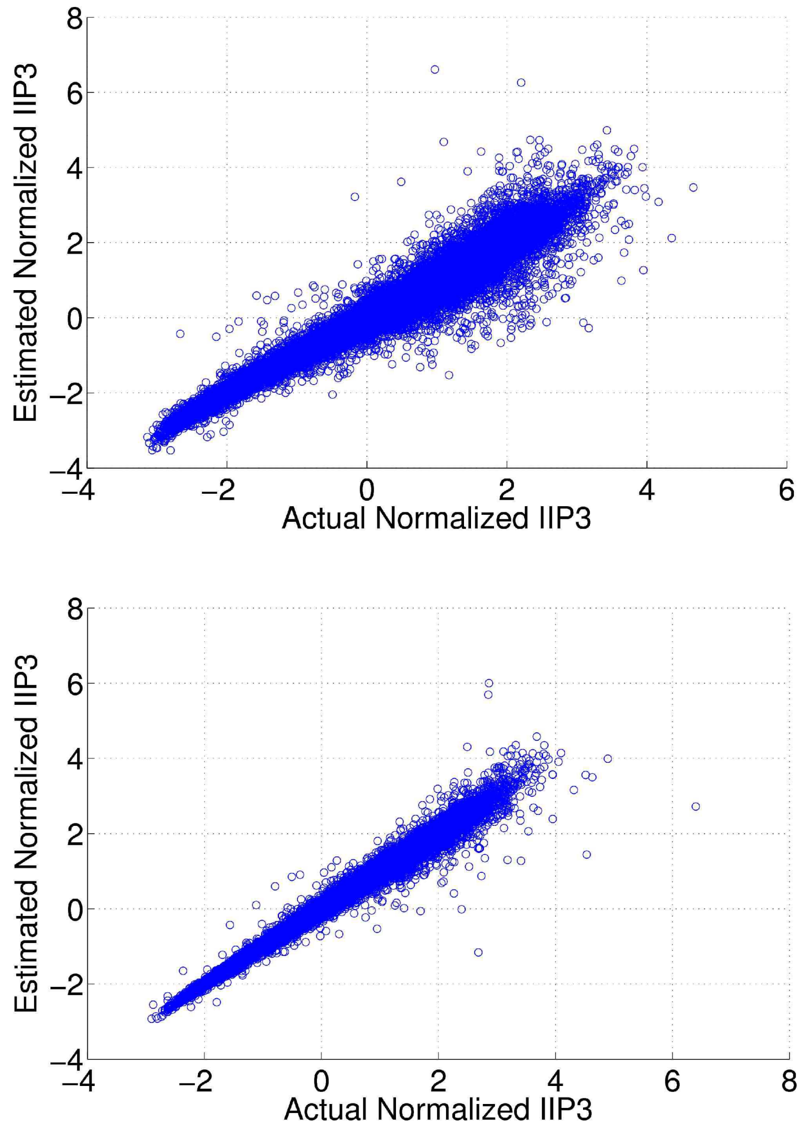


Fig. 20. Input space needs to be properly sampled to achieve a good fit. In (a), the model response is poor due to limited size of the training set. However, goodness of fit can be improved by sampling from wider distribution for the same size of training instances.

stances are generated and simulated in the Monte Carlo approach. For our method, we first simulate 5k samples, which is used for training, and simulate only filtered important

Fig. 21. Experimental circuit: LNA-Mixer

	DPPM			DPPM-σ		
	MC	Proposed	*[]	MC	Proposed	*[]
TL #1	1290	1220	2880	114	110	170
TL #2	3030	3030	8820	174	174	300
TL #3	3840	3140	4370	196	177	208
TL #4	1670	1640	1660	130	128	129
TL #5	5430	5390	5386	233	232	714
TL #6	2880	2760	4090	170	166	202

*[]= [Stratigopoulos *et al.*(2009b)]

Table 2. Estimation result and their corresponding standard deviation (TL=Test List)

samples. For fair comparison, we used the same data set that is used for Monte Carlo simulation as initial sample set. Simulations of method in [Stratigopoulos *et al.*(2009b)] is done by using 5k instances for training and regenerating 100k instances using the training information. Since we target DPPM estimation in this work, we compare DPPM estimation performance of the three methods. DPPM depends on the selected subset of tests. We randomly generated 7 test list selecting from the 10 specifications of the circuit. Simulation results are shown in Table 2. Evaluation results of estimated DPPM values and their corresponding standard deviation are shown for 6 different test lists in separate row. Results show that the proposed method is able to yield very close results compared to the Monte Carlo method, which we use as the baseline of evaluating our method. Compared to the MC method estimated level is acceptable and has an acceptable error. The variance of the proposed technique is either smaller or equal to the variance of the baseline MC simulation. The results of [Stratigopoulos *et al.*(2009b)] indicates that the method yields good estimates for some of the test lists but not for all of them, therefore is not consistent.

Table 3 shows the computation time improvement of our technique compared with the baseline Monte-Carlo method. Number of required simulations after filtering are listed in the 2nd and 3rd column in Table 3 for MC and the proposed method. Results show that 10x to 25x reduction in number of simulations is achieved compared with MC method. Total simulation time, including the training time is listed in the 4th and 5th column of the table. Achieved overall simulation time reduction is approximately 18 fold according to the results. Note that the 100k MC simulations have a DPPM variation on the order of 10^2. In order to achieve higher accuracy, several million simulations may be necessary. Also note that the majority of simulation time for our technique is the training phase. For instance, for 1 million MC samples, we can still use the ANN trained by 5k samples, and the number

	# of simulations		Total Simulation Time	
	MC	Proposed	MC [Hr]	Proposed [Hr]
TL #1	100k	424	1100	60
TL #2	100k	743	1100	63
TL #3	100k	958	1100	66
TL #4	100k	216	1100	57
TL #5	100k	574	1100	61
TL #6	100k	521	1100	61

(TL=Test List)

Table 3. The number of required simulations, and total simulation time for each method.

of required simulations will scale up by 10, which would still be less than 1k. As a result, total computation time savings from our technique would be nearly 200x.

2.7. *Summary*

We presented an efficient methodology to estimate DPPM in considerably shorter time. A model based approach is employed to successfully select only the information containing device samples for simulation. In this two step approach, we first selected potential DPPM contributors through a model based filter and then simulated the filtered important samples. We also addressed the issue of errors in sample selection due to model imperfection and robust training. Our proposed method outperformed the Monte Carlo method by 10x-25x for the experimental circuit used in this work.

The proposed work is versatile in the sense that it is not circuit specific. The method is very effective in reducing the simulation time because contrary to the generic Monte Carlo

method, our technique is focused on a particular statistical parameter. The idea can be easily generalized to estimate statistical parameters other than DPPM.

The method is compatible with the widely adopted MC method, it requires only insertion of the filtering step without changing the simulation setup.

3. Defect Oriented Test Selection

Analog fault modeling (AFM) provides a quantitative measure of quality and insight into defective device behavior. However, the high computational burden typically associated with fault simulation makes it unappealing for industrial applications. We propose an efficient methodology to reduce computational burden of the AFM method by exploiting the hierarchical nature of process variation. We apply the proposed methodology on an industrial SerDes TX Driver circuit and achieve 98% simulation time reduction. We quantify defect impact with a defect severity measure

3.1. *Analog Fault Modeling Approach*

We focus on structural defects and use resistive opens and shorts which are commonly used defect models. The inclusion of frequency related defect models as discussed in [Acar and Ozev(2008)] was deemed unnecessary for this evaluation. Since defect size can vary, we investigate multiple resistance values for each defect location to avoid incorrect conclusions on defect coverage.

3.1.1. *Defect Oriented Simulation Flow.* Defect oriented simulation can be separated into four steps: defect list generation, defect injection, simulation, and defect assessment.

1. Defect list is generated examining the schematic or the layout of the circuit. This list includes possible defects generated during manufacturing process due to various defect mechanisms, such as extra metal deposition.

2. Assuming single defect model, defects are injected to the circuit one at a time.

3. Defective circuits are simulated.

4. Performance parameters are analyzed for specification or test result violations and a coverage table is generated.

For this study we generated the defect list via the schematic method. Generating a defect list via a layout analysis [Meixner and Maly(1991)] results in the most accurate defect list and offers a ranking of defects based process defect statistics. However as it is an established technique we wanted to focus on the simulation challenges of including process variation. In addition for test coverage analysis waiting for a layout does not enable improvement of test coverage with early feedback. The results from performing AFM at the schematic level can be used to improve defect robustness.

3.1.2. *Process Variation Model.* Although circuits are designed to tolerate process variation, when combined with defects, process variation can result in unpredictable consequences. Therefore, it is necessary to incorporate the effect of process variation.

Process variation consists of several layers. We group these layers into high-level and low-level variation such that we separate correlated and uncorrelated variation. We define high level variation as the overall effect of lot-to-lot, wafer-to-wafer, and die-to-die variation. Low-level variation is defined as the effect of WID (aka mismatch) variation. High-level variation is typically higher and varies parameters (e.g. transistor Vt) of the same nature at the same rate; hence, it is common mode variation. Low-level variation can be considered as independent variation.

Splitting the variation enables us to conduct analysis in two consecutive steps. First, the circuit is subjected to high-level variation through sampling device parameters from the statistical distribution model of high level variation. In this work, a skew based statistical model is employed. Conceptual representation of the model is provided in Figure 22. The figure shows fast and slow corners of the silicon for important process level parameters.

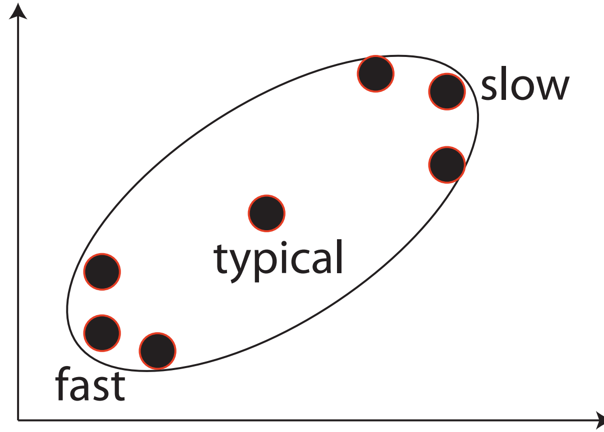


Fig. 22. Skews Model for Process Variation

High level variation is represented with a number of skew points that bound the most likely region of the distribution. In the second step, Monte Carlo simulation on selected transistors is conducted for each skew point to simulate for the effect of WID variation.

Considering that there may be many possible defects and simulating for each defective circuit for process variation requires sampling a large number of device instances, it is not surprising that defect oriented approach is computationally expensive. In this work, we show how we can mitigate the computation burden and reduce simulation time substantially by exploiting several observations. These will be described in the next subsection.

3.2. *Computationally Efficient Defect Simulation Strategy*

Simulation process of our approach consists of three consecutive steps;

1. Simulating the typical behavior of defective circuits

2. Simulating high level variation for each defective circuit

3. Simulating low level variation for each defective circuit

The third step is computationally the most expensive. Therefore, skipping this step reduces the simulation time appreciably. To this end, we make use of four observations.

- Impact of high level variation differs for fault-free and faulty devices. Process variation for defective devices cannot be predicted from the fault free simulation results. We illustrate this observation through a sample defect. Figure 23 shows the response of a short defect we have analyzed. X-axis shows the resistance values of the defect models ranging from 1 to 5k ohms. Y-axis shows the response in terms of one sample performance parameters. Red dashed lines indicate the range of fault free response in presence of process variation. Vertical blue bars show the range of defective response in presence of process variation for six different defect models. Analysis of several defect responses reveals that response variation depends on the defect model and it cannot be estimated from fault free response. Therefore, it is not possible to skip the second step of simulations.

- Hierarchical process model reduces computational burden. Process variation is typically emulated by sampling from a parametric distribution and this approach requires a large number of device instances to be sampled. Since skew parameters are much smaller in number, an exhaustive evaluation of the high level process space is possible by separating the WID variation from high level variation. The total number of simulated skews is 7.

- The effect of WID variation is similar for faulty and fault free circuits. Although high-level variation differs for fault-free circuits and faulty circuits, WID variation results in similar tolerance range for both fault free and faulty circuits (see Figure 24).

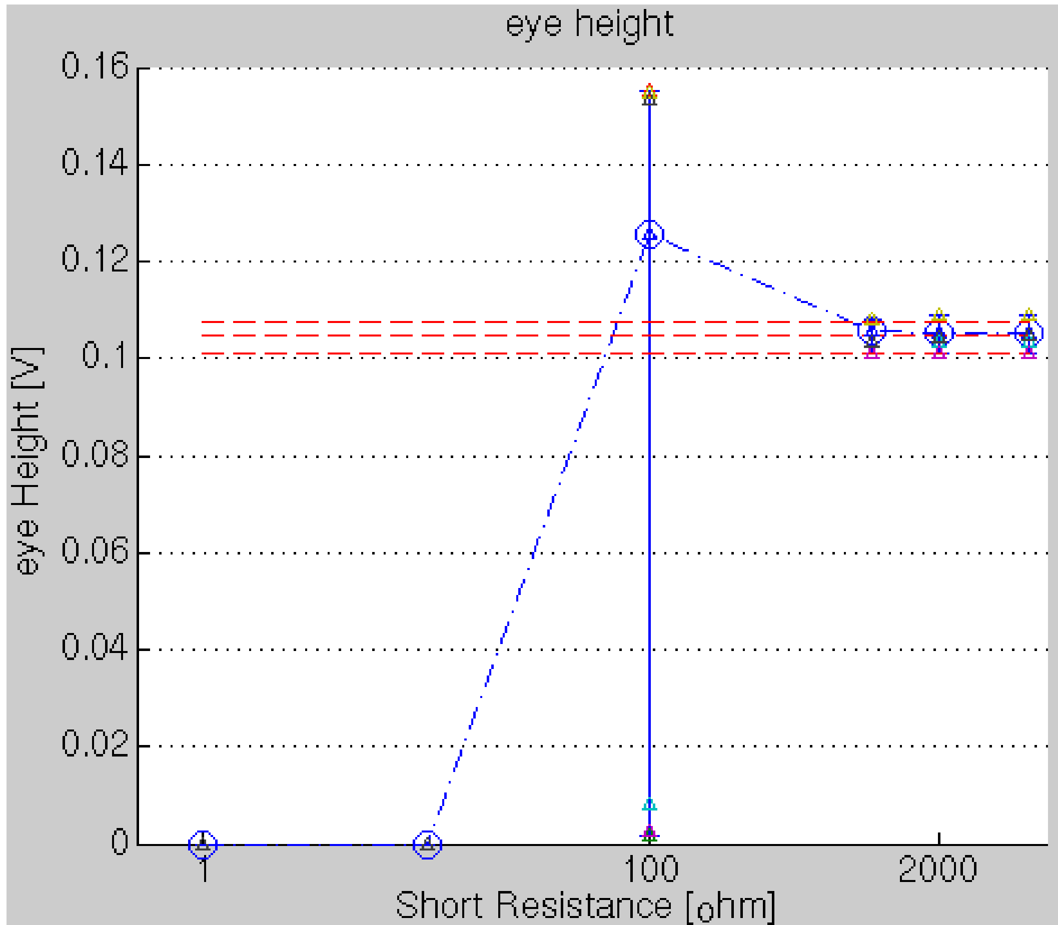


Fig. 23. Performance parameters in presence of a short between the output node and ground

Therefore, within die variation level obtained for fault-free circuits can often be used as a guide to estimate WID variation for defective circuits. In cases where WID variation can be estimated with high confidence, mismatch simulation can be avoided. Therefore, this observation enables us to reduce to number of 3rd step simulations.

• Analysis of the circuit architecture enables further reduction in simulation time. Defect extraction effort and simulation time can be significantly reduced by isolating sub-blocks from the circuit and simulating them.

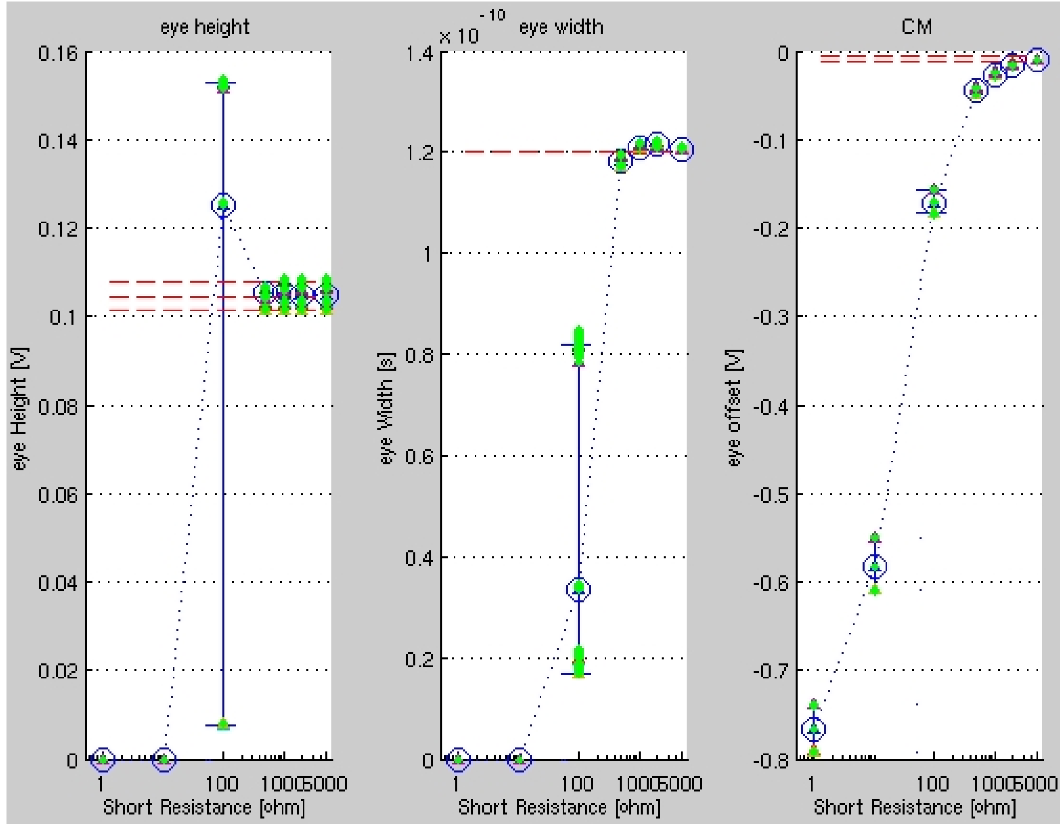


Fig. 24. WID impact: short between output node and ground

3.3. *Case Study: PCI-Express TX Driver*

We demonstrate our feasible AFM method on a PCI- Express TX analog front-end circuit in a 65nm process.

3.3.1. *Circuit Background:PCI-Express TX Driver Circuit.* We chose to analyze the analog driver circuit, which is the outmost circuit interfacing the channel. The driver circuit is a 5 bit source series terminated digital to analog converter [Menolfi *et al.*(2007)]. High level diagram of the driver is illustrated in Figure 25. Each incoming bit is connected to a set of identical cell circuits. Each bit is connected to 2n cells, where n is the significance number of the bits. The driver generates signal levels proportional to the number of the active cells. Note this is design for a 65nm process.

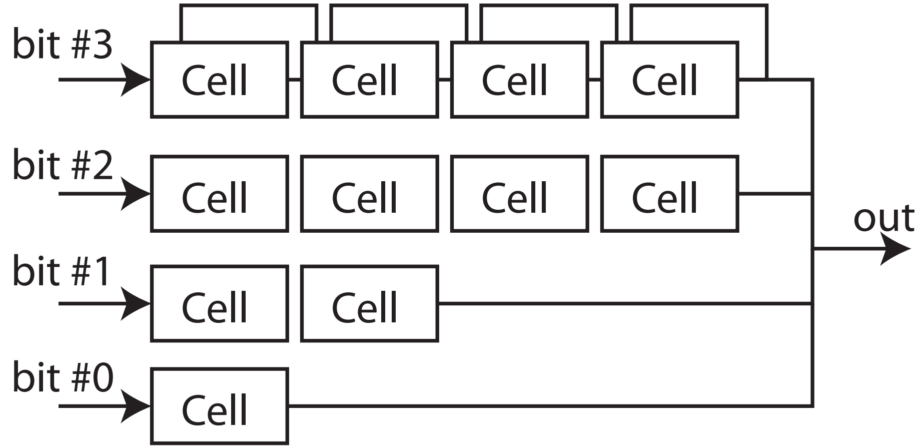


Fig. 25. High-level representation of the Tx driver

3.3.2. *Simulation Setup.* High level representation of the simulation setup is shown in Figure 26. We included the test environment model to obtain realistic responses. Pseudo random input is applied at the input, coded according to the PCI express standard. We evaluated the TX Driver performance with respect to three commonly used test parameters for high speed links: eye height, eye width, and eye offset [Mak *et al.*(2004)]. Eye height and width measure the vertical and horizontal opening of the eye diagram, while eye offset measures vertical shift of the eye. Eye height is typically determined via voltage margining [Meixner *et al.*(2008)] method by introducing a common mode shift until the signal becomes undetectable. Similarly, eye width is measured through time margining; introducing a shift in time domain. Eye offset can be determined by computing the average integral of the differential input.

3.4. *Defect List Generation and Defect Equality*

Defect list is generated by analyzing the circuit to find possible defect locations. We exploit the modularity of the driver to reduce the defect list extraction effort. Since all cells are identical, examining only one cell suffices to generate the complete defect list. The

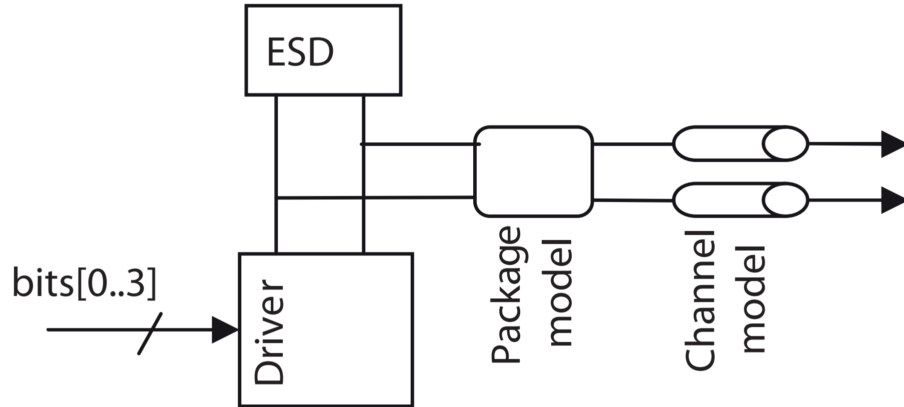


Fig. 26. Simulation setup

generated defect list can be duplicated for the other cell circuits in the driver. Also, the cell circuit has a symmetric structure enabling further effort saving through examining only one half of the cell circuit. However, identical defects in various building blocks do not necessarily result in identical behavior.

Even though all cells are identical, their response may differ due to differing input bit patterns. We group the cells with identical inputs. Simulation for each group is necessary to obtain an accurate overall response. For this design, cells are organized into 4 groups depending on input bits connected. A further reduction can be obtained through symmetric architecture of the cells. The overall simulation savings using architectural information are discussed in the results section.

A defect list of 17 shorts and 15 opens is generated through examining the cell schematic using practical heuristics for opens and shorts as follows:

- All nodes can be shorted to either VCC or Vss
- For all Transistors Gate to Drain, Gate to Source and Drain to source shorts included
- Opens introduced at all likely junctions

3.5. *Defect Injection*

Defect-oriented simulation requires knowledge on the circuit level model of the defects. Shorts and Opens are both modeled with resistors. Traditionally, short defects are assigned 1 ohm and open defects are assigned a large 1M ohms. However, depending on the defect mechanism and the defect size, defect models can assume a wide range of resistance values. In this work, a realistic range of defect resistances for the 65nm process was chosen for simulation as follows:

- Shorts: 1, 100, 1K, 2k, 5K Ohms
- Opens: 1k, 2K, 5K, 10K, 100K, Infinity ohms.

3.6. *Simulation Flow*

Figure 27 shows the simulation flow. We start by simulating the fault-free circuit which yields the fault-free response in the presence of high-level and low-level process variation. These results serve as a reference to assess defective responses.

Defective circuits go through a similar flow; however, we apply a pruning method to reduce simulation time. First two steps of the simulation are necessary for all defects. The third step may be dropped in cases where WID variation of a defective device is guaranteed not to change the pass/fail criteria. The proposed pruning algorithm decides whether a defective device needs to be simulated for WID variation.

3.7. *Pruning Algorithm*

Once we obtain the typical response and high-level variation response of defective circuits through first two simulation steps, we can estimate worst case response of WID varia-

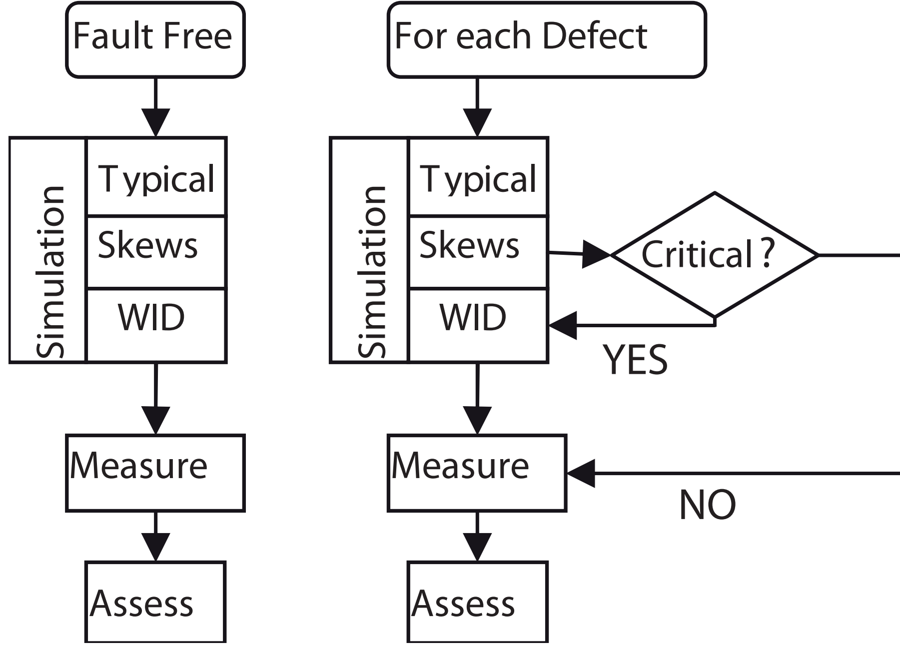


Fig. 27. Simulation Flow

tion utilizing WID variation results obtained for the fault-free circuit. Worst case variation limits are defined using the equations listed below.

$$pp_{i,min} = min(pp_{i,skew\{j\}} - 6 \cdot std(pp_{i,FF,WID})) \quad (2.15)$$

$$pp_{i,max} = max(pp_{i,skew\{j\}} + 6 \cdot std(pp_{i,FF,WID})) \quad (2.16)$$

$$spec_{i,min} < pp_{i,min} < pp_{i,max} < spec_{i,max} \quad (2.17)$$

$pp_{i,skewj}$is the ith performance parameter for j^{th} skew, where j is the skew index. $Std(pp_{FF,WID})$ is the standard deviation of the response of ith performance parameter for WID variation. Since WID variation does not change considerably for fault-free and defective circuits (observation #3), we use fault-free WID variation amount as a guide to estimate the worst case scenario for defective responses. To increase the confidence of

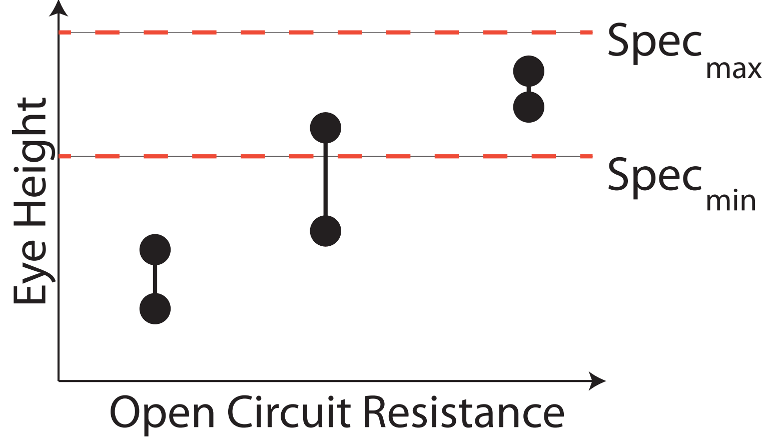


Fig. 28. Defect pruning critical defect

the approach, we chose 6σ window. Equations (2.15) and (2.16) yield a minimum and a maximum worst case point for the performance parameter of defective circuit.

We define a defect as critical if it does not satisfy Equation (2.17) for any of the defect models. Critical defect concept can be explained via Figure 28 which illustrates the response obtained for a critical defect for 3 defect models. Red dashed lines show the acceptable region confined by minimum and maximum specification limits. Vertical lines show the worst case response range, where solid circles at the bottom and the top of the lines are $pp_{i,min}$ and $pp_{i,max}$ points respectively. According to our pruning strategy, if any of the behaviors fall outside of the acceptable region, a defect is considered as critical. In Figure (28), defective response violates the criteria given in Equation (2.17) for the first two defect resistance values; hence it is identified as critical.

This algorithm identifies the defects that are susceptible to WID variation. For example, the response range for the second defect model in Figure 28 intersects with one of the specification limits. Hence, there may be a set of device instances for this particular defect, whose performance parameters fall very close to the boundary. These instances can easily

be pushed to the other side to the limit by WID variation. Pruning algorithm guarantees to capture those cases in the specified defect resistance range.

3.8. *Fault Coverage Assessment*

Fault coverage assessment and pruning algorithm requires an acceptable region definition of the performance parameters. These specifications are given by the designer. For the circuit in this study, the specifications are as follows:

- Eye Height (min): 80mV
- Eye Width (min): 118ps
- Offset: +/- 200mV

Depending on the fault location and the defect size, each fault may not result in a specification violation. Our goal is to compute the probability of specification of violation for each fault and the probability of detecting this violation by one of the three tests defined above. We use these two measures to calculate fault coverage. Fault coverage is defined below as:

$$
\begin{aligned}
FP_{Fi} &= Pr\begin{pmatrix}\text{device with fault } F_i\\ \text{fails at least one spec}\end{pmatrix}\\
DP_{F_i}^{T_j} &= Pr\begin{pmatrix}\text{device with fault}\\ F_i \text{ fails test } T_j\end{pmatrix}\\
FaultCoverage_{F_i}^{T_j} &= Pr\begin{pmatrix}\text{device with fault } F_i\\ \text{fails test } T_j \text{and at}\\ \text{least one specification}\end{pmatrix}\\
FaultCoverage_{F_i}^{T_j} &= 100\frac{DP_{F_i}^{T_j}}{FP_{F_i}} \qquad (2.18)
\end{aligned}
$$

The above equation is valid if the test (T_j) set consists of direct specification based tests.

3.9. ***Analysis of Design Robustness with Respect to Defects***

The impact of defects on performance may be substantially different; some defects degrade the performance severely and deserve more attention. A priori knowledge of such sensitive defect locations can be used by designers to build defect-robust circuits. For instance, if the drain of a transistor is sensitive to an open defect, placing multiple connectors (vias) at the drain will make the design more robust against this defect.

In order to assess the impact level of a defect, we define the defect severity (DS) metric as described by Equation (2.19).

$$
DS(R) = \begin{cases} \frac{pp_{i,typ}(R)-pp_{FF,typ}}{spec_{max}-pp_{FF,typ}} & \text{, if } pp_{i,typ} > pp_{FF,typ}\\ \frac{pp_{i,typ}(R)-pp_{FF,typ}}{pp_{FF,typ}-spec_{min}} & \text{, else}\end{cases} \qquad (2.19)
$$

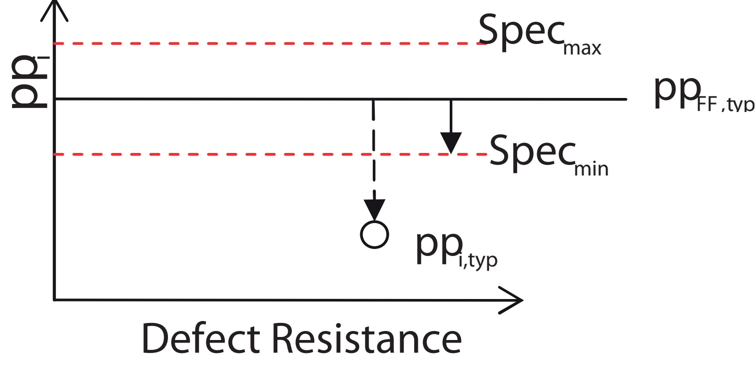


Fig. 29. Normalized defect severity indicates the defect impact

where, $pp_{i,typ}$ is the ith performance parameter for of the typical skew and $pp_{FF,typ}$ is performance parameter of the fault-free response for typical skew. This formulation defines the deviation of the response from the nominal in terms of allowed deviation. Defect severity metric is illustrated in Figure 29. The vertical dashed line shows the deviation of a performance parameter for a defect with a particular resistance model. DS metric is simply the ratio of the length of the vertical dashed line to the length of the vertical solid line for this particular example. DS is a function of defect resistance value.

We define another metric, DSS (Defect Severity Score), to estimate the severity conditioned on defect resistance distribution in Equation (2.19). This metric incorporates the statistical distribution of defect resistance value.

$$DSS = \int_0^\infty DS(R)pdf(R)dR \tag{2.20}$$

where $Pdf(R)$ is the probability distribution function of defect resistance distribution. DSS indicates the severity given a particular defect takes place in the device. However, the probability of that particular defect may be low, which would obviate the attempt to alter the design to make circuit robust for that particular defect. Or the probability may be high

	Time [Hr]
Sim. Step #1, Typical	12
Sim. Step #2, Process skew	96
Sim. Step #3, WID	1296

Table 4. Simulation time

which would make it important to alter the design for improved robust operation. Expected value of DSS can be estimated by incorporating probabilities of the defect occurrences through weighing DSSs with defect occurrence probabilities. Expected defect severity score ($EDSS$) metric is defined in Equation (2.20).

$$EDSS_k \quad = \quad Pr\left(Defect_k\right) DSS_k \tag{2.21}$$

where, k is the defect index and $Pr(Defect_k)$ is the probability of occurrence of kth defect. Ordering the defects with respect to EDSS enables us to assess the realistic impact of the defects and devote resources such as redesign effort and die area more efficiently.

3.10. ***Results***

The proposed simulation flow was applied to the PCI express driver circuit. Simulations were run on 3GHz Quad core machines using multiple threads. We report simulation time in terms of equivalent CPU time of a single core machine. Table (4) lists the duration of each simulation step. In the first two steps, all defects are simulated, while in last step only the 18 defects that are identified as critical in the pruning step are simulated. The most significant contributor is the WID variation simulation.

	Final Simulated	No pruning	No defect equality
Time [Hr]	1404	9468	71010
Saving		85.2%	98%

Table 5. Simulation time savings

Results in Table (5) show that simulation time without pruning is 9468 CPU hours, while it is 1404 CPU hours with the proposed pruning algorithm, saving 85.2% of the simulation time.

Defect equality approach enables simulation time reduction as well. 4 out of 15 cells are simulated due to the identical structure of the cells, and only one half of the defects per cell are injected thanks to symmetry. Hence, the overall saving of the pruning and defect equality approach is 98%. Without defect equality, the required simulation time is 71010 (extrapolated) hours.

3.10.1. *Fail Probability and Per-Test Fault Coverage.* 32 defects for 4 different cells and 2 defects that are common to all cells are simulated for 6 defect models. Out of 130 defect types, only 18 of them proved to be critical. This suggests that not all defects result in a specification violation. Defect coverage table is shown in Table (6). The leftmost column shows in which cell the defects are located. The second column indicates the type of the defects, the 3rd column shows fail probability, and the rest of the columns show the coverage of faults for individual performance parameters.

Results show that faults are fully covered by eye height and eye width test and eye offset test does not improve the coverage. This information enables us to optimize test

Bit#	type	FP_{F_i}	Fault Coverage		
			Eye Height [%]	Eye Width [%]	Eye offset [%]
all	short	0.50	80.96	100	66.66
all	short	0.50	42.86	100	19.04
2	short	0.33	100	0	0
2	short	0.50	100	0	0
2	short	0.33	100	21.42	0
2	short	0.50	100	0	0
2	short	0.50	100	33.34	0
2	short	0.36	100	0	0
2	short	0.67	100	0	0
2	short	0.50	100	33.34	0
2	open	0.67	100	21.43	0
2	open	0.74	100	12.90	0
2	open	0.26	100	0	0
2	open	0.67	100	21.43	0
3	short	0.14	0	100	0
3	short	0.33	0	100	0
4	short	0.14	0	100	0
4	short	0.17	0	100	0

Table 6. Fail probability and fault coverage table

process by dropping redundant tests and ordering tests to reduce the expected test time. For instance, we can drop eye offset test, and scheduling eye height test before eye width test will reduce the expected test time for fails. This assumes defects are equally likely. For this case study the analysis is straightforward to conclude by manual observation for this case study, algorithms can be developed for test optimization when the number of specifications is much higher.

3.10.2. *Analysis of Design Robustness.* Based on the fault simulation results, only 18 out of 130 defects result in failure. Therefore, schematic based analysis showed that the circuit can be considered robust with respect to most of the defects. In order to further improve robustness, we only need to focus on these 18 sensitive defects.

Robust design techniques can be applied at the layout level by locally laying devices out and routing wires to reduce the probability of a failure at a potential cost of area. Defect severity measure enables one to optimize the cost of this effort by prioritizing via defect impact.

Equations (2.18)-(2.20) can be used to obtain the severity of the defects to improve the design for defect robust operation. Defect resistance distribution, $pdf(R)$, is considered uniform for 65nm process based upon previous internal analysis. In this work we assumed equal occurrence probability for all defects (due to schematic based defect generation). These two assumptions reduce equations (2.20) and (2.21) to equation (2.19). Hence, we used only equation (2.19) to evaluate the severity of the defects.

Figure (30) shows defect severity plot of the 18 critical defects. X-axis shows the number each defect and y-axis shows severity in terms of DS parameter. The first 14 of the bars are for short defects and the last 4 bars are for open defects. The contribution of each defect

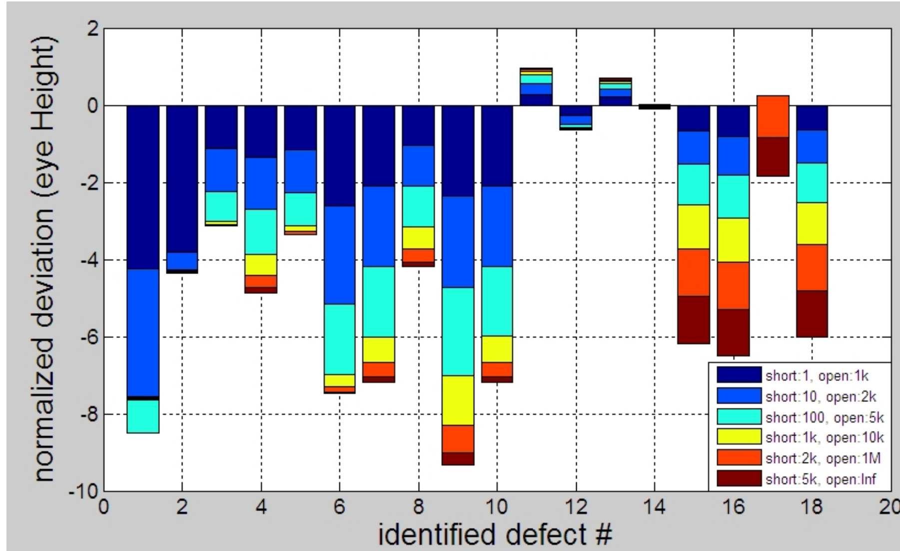


Fig. 30. Defect severity

resistance model is represented with a different color. Height of the bars indicates which defect is more important, providing useful information to improve defect robustness.

According to these results, the most influential defect is the 9th, therefore it should be addressed first. The 1st and the 6th defects are next in the priority list. Defect priority list enables the layout engineer to devote the limited chip area to important defects more efficiently.

3.11. *Summary*

We presented an efficient implementation methodology for AFM technique on an in industrial SerDes driver circuit using an industry process and its associated process variation. Results show 98% simulation time reduction compared to the traditional AFM implementation. Given the trend in VLSI design to rely on IP blocks that would be used for multiple products, the simulation investment in assessing an IP's analog fault coverage is worthwhile. The AFM assessment permitted a realistic assessment of the manufacturing

tests. The resulting 18 defective circuits had any significant impact and that eye height and width together provide complete fault coverage. In addition, we suggested improving design robustness by defining a defect severity measure and assessing it on all 18 defects. This measurement enabled us to rank their impact and hence, to improve the circuit layout.

There exist several directions for future work. The most immediate is to follow our case study with actual silicon results. A simple experiment would be to assess the overlap between eye height and width measurements. Next would be to apply this methodology to more complex analog circuits e.g. a clock data recovery circuit. Out of the four basic observations that we utilize to make AFM efficient, the first two are general and apply to all circuits. Utilization of last two observations which employ circuit specific information (symmetry and WID variation) while not necessary, will significantly help to boost the efficiency as demonstrated in this paper. As analog circuits often have multiple identical elements and differential circuits are required in SerDes like devices it is not unrealistic to presume these can be used to advantage. Finally the focus of this work has been on analog fault coverage so another direction to apply this work is to the much harder problem of analog yield prediction.

CHAPTER 3

Post-Silicon Test Strategies: High Volume Test Optimization

1. Per-device Adaptive Test for Analog/RF Circuits

We present an adaptive test flow for mixed-signal circuits that aims at optimizing the test set no a per-device basis so that more test resources can be devoted to marginal devices whereas devices that are not marginal are passed with less testing. Cumulative statistics of the process are monitored using a differential entropy based approach and updated only when necessary. Thus, process shift is captured and continuously incorporated into the analysis. We also include provisions to identify potentially defective devices and test them more extensively since these devices do not conform to learned collective information. We conduct experiments on an LNA circuit in simulations and apply our techniques to production data of two distinct industrial circuits. Both the simulation results and the results on large-scale production data show that adaptive test provides the best trade-off between test time and test quality as measured in terms of defective parts per million.

1.1. *Methodology*

In [Yilmaz and Ozev(2008), Yilmaz and Ozev(2009a)] we have shown that the performance limitation of static compaction methods can be removed through adaptive testing, where information obtained during the test phase can be incorporated and used to predict the behavior of each individual device under test (DUT). If measurement results of conducted tests can give an insight about the DUT, we can adapt testing process to the DUT and extract the most information in fewer executed tests. In [Yilmaz and Ozev(2009a)] we have employed a simple correlator that relates the value of a measured specification to the pass probability of thus-far unmeasured specifications. This correlation is used to

update the pass probability of each unmeasured specification and decide which tests can be safely skipped. In this work, we develop several new methodologies to better estimate the pass probability of unmeasured specifications. Moreover, we provide mechanisms to identify potentially defective devices and subject them to more exhaustive testing.

1.2. *Adaptive Test Problem Formulation*

First, we formulate the adaptive test problem. The relation between circuit parameters and specification parameters can be represented using a set of equations, as shown in equation(3.1).

$$\begin{aligned} s_1 &= f_1(x_1, x_2, \ldots, x_N) \\ s_2 &= f_2(x_1, x_2, \ldots, x_N) \\ &\vdots \\ s_M &= f_M(x_1, x_2, \ldots, x_N) \end{aligned} \tag{3.1}$$

where, $f_i(\mathbf{x})'s$ are system functions that relate circuit parameters, such as resistance, capacitance, width/length denoted with x to specification domain parameters, s_i. Each specification has an acceptable region in order to identify defective devices before shipping. Measurement results (s_i) of i^{th}test must satisfy the condition given in equation (3.2), where $spec_i^-$ and $spec_i^+$ are lower and upper specification limits respectively.

$$spec_i^- < s_i < spec_i^+ \tag{3.2}$$

Since circuit parameters vary due to the process variation, input parameters of the system function are probabilistic.

Therefore, output parameters are also probabilistic. Given the distribution of the input parameters, it is possible to obtain the joint distribution of the specification parameters using the system transfer function. Once we obtain the joint probability distribution of specification parameters, we can compute pass probability of each DUT using equation given in equation (3.3).

$$p(pass) = \int \cdots \int_{spec_i^-}^{spec_i^+} pdf(\mathbf{s}) d\mathbf{s} \tag{3.3}$$

If we do not have information about the DUT, the best we can do is to predict its passing/failing probability using cumulative information. However, after each test, information is revealed about the behavior of the DUT. Conducting a test on the DUT yields one of the specification parameters and therefore reduces the degree of integration. If there is a correlation between the measured specification parameter and unmeasured ones, probability distribution function is altered resulting in a change in passing probability. Updating the joint distribution of specification parameters can be viewed as a learning process. Hence, we learn more about the DUT after each measurement.

The objective of adaptive testing is to use this information to adapt to the behavior of the DUT and skip tests that are guaranteed to pass with high confidence. If there is correlation between specifications, after several measurements, pass probability of unmeasured specifications should either reduce or increase. Hence, we can skip the tests that we are confident will pass without jeopardizing test quality while reducing test time.

In reality, however, most of the above mentioned model parameters such as system transfer function, distribution profiles of circuit components and therefore joint distribution of specification parameters are not known. However the JPDF can still be generated using a training set and kernel based estimation methods [Stratigopoulos *et al.*(2009b),Scott(2008)].

1.3. *Background: Kernel Based Estimation*

In this work, we employ a kernel based probability distribution estimation approach for our adaptive testing technique. Kernel based estimation enables us to capture the correlations between specification parameters and update the JPDF after each measurement. Moreover, generated PDF provides an inherent mechanism to decide which tests to conduct. In this section, preliminary information related to kernel based estimation is provided.

Kernel function is defined in equation (3.4). Among several well known kernels, such as Epanechnikov, Gaussian, and Bi-weigh, we chose to use Gaussian distribution as kernel in order to benefit from the large literature devoted to its analysis. For our purpose, selection of a kernel was not a critical concern; we obtained similar results with different kernels. Equation (3.5) is the PDF function estimated using kernels, where $\mathbf{S}_i$ is a vector containing specification parameters of i^{th}device in the training set, h is the kernel width, w_i is kernel weight, n is the size of training set and M is the number of specifications.

$$\int_{-\infty}^{\infty} K(x)dx = 1 \tag{3.4}$$

$$\hat{pdf}(\mathbf{s}) = \frac{1}{n\prod h_j} \sum_{i=1}^{n} w_i \prod_{j}^{M} K\left(\frac{\mathrm{s_j} - \mathbf{S}_{i,j}}{h_j}\right) \tag{3.5}$$

Kernel based PDF estimation technique simply superimposes kernels ($K(x)$) on each observation in training set (S_i). It can be viewed as convolution of the training set data with the kernel. Parameter h sets the width of the kernel, hence we can adjust the width in order to achieve the best fit. Equation (3.6) specifies the optimum width for Gaussian kernel in order to minimize the mean integrated squared error (MISE), where σ_j is standard deviation of the jth specification parameters estimated using standard deviation estimator, and d is the number of dimensions (non measured parameters).

$$h_j = \left(\frac{4}{d+2}\right)^{\frac{1}{d+4}} \sigma_j n^{-\frac{1}{d+4}} \quad (3.6)$$

1.4. *Kernel Based Adaptive Test Approach*

The basic principle of our approach is to incorporate on-line measurement data in conjunction with the collective data in order to achieve the best test quality versus test time trade-off possible. For that purpose, we first need to obtain the cumulative information of the device ensemble. We collect this information in the static phase of our method through exhaustive testing of a training set of samples. Training sample set is representative of the devices produced with the same process, hence, it is the characterization data of the process. We represent measurement results of training sample set with $\mathbf{S}_i$, where S is a vector of specification parameters, and i is the index of the device in the training set sample.

The adaptive test method presented in this work starts with an initial, ordered test list and adaptively selects which tests to conduct. Selection of the initial test list, $\{T_j\}_{j=1}^{M}$, is explained in section 1.6. Generation of this pre-ordered test list is performed in static phase.

Figure 31 shows the flow of our proposed adaptive test method. Static operations that are mentioned above are illustrated at the upper part of the diagram, while the adaptive part is located below the dashed lines of the diagram. The adaptive testing process runs for each device, D_k, in the production line. When the method is executed, for each device (D_k), the first test, T_1, in the test list is conducted and the specification parameter for that test is is recorded (s_j). D_k's measurement response is compared to specification limits and the DUT is identified as a fail if specification limits are violated. If the specification parameter is in the acceptable limits then the algorithm proceeds with update procedure, where joint

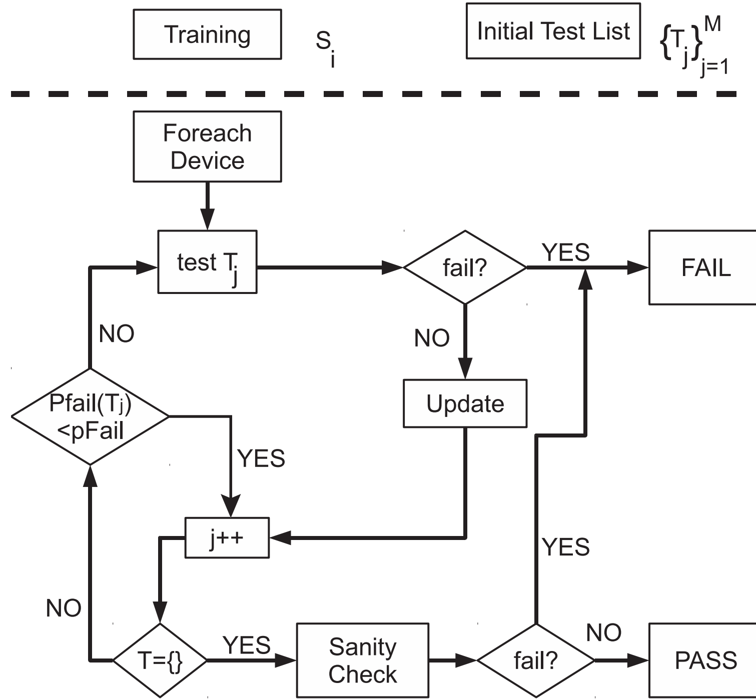


Fig. 31. Adaptive testing flow diagram. The flow adapts test procedure per-device depending on the particular characteristics of the DUT.

PDF ($\hat{pdf}(s)$) of unmeasured specifications is updated using the measured parameter value. In the following step, the next test is selected and the JPDF is checked if the failure probability of the related specification parameter is greater than a threshold value, $pFail$. If the probability of failure of test T_j is less than that threshold, it is skipped and this procedure repeats until a test that is more likely to fail than that threshold value is found. If such a test is determined, the algorithm proceeds to the measurement step, and this procedure is continued until the test list is exhausted or any of the tests fail. In order to identify a device as a good device, either test list must be exhausted or failing probabilities of all unmeasured tests be less than the threshold of failing probability ($pFail$).

1.4.1. *Joint PDF Update.* Estimate of the JPDF of the unmeasured specifications can be obtained using equation (3.5). In the update step, measurements results of the conducted test is used to update the JPDF. Suppose that the analytical form of JPDF is known, then the only thing we need to do would be to plug the measurement result into JPDF and integrate with one less dimension in order to get the fail probability. However, the analytical form of the JPDF is not known in a real life scenario. Hence, we approximate the conditional JPDF through properly adjusting weighing coefficients of kernels in the JPDF estimation equation (3.5). We penalize the kernels proportional to their distance from the measured value by reducing their weighing coefficient. This enables us to achieve localization around the measurement point and therefore adapt to the behavior of the DUT. Weight penalization is performed using the kernel function, however, its window width is reduced such that information in the vicinity of the measurement result is treated as valuable. Equations (3.7) and (3.8) show how penalization is applied in the estimator function.

$$\hat{pdf}(s_i) = \frac{1}{(\sum w_i') \prod h_j} \sum_i w_i' \prod_{s_j \in T} K\left(\frac{s_j - S_{i,j}}{h_j}\right) \tag{3.7}$$

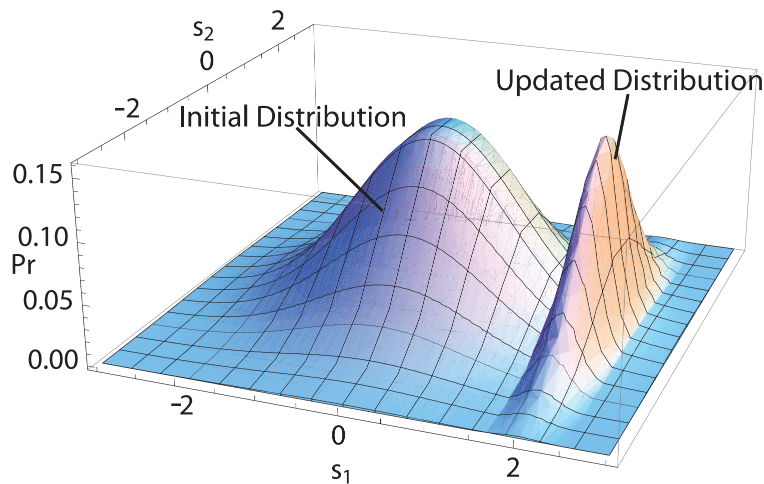
$$w_i' = \left[\prod_{m_j \in T'} K\left(\frac{m_j - S_{i,j}}{\alpha h_j}\right)\right] w_i \tag{3.8}$$

where, w_i' is updated weight of ith kernel, T' is the set of conducted tests, and α is kernel penalization coefficient. Once the penalized weights (w_i') are calculated, they are normalized such that they sum up to n. A simple example for JPDF update is shown in Figure 32(a). In this example, the large distribution located in the middle of the plot is initial JPDF, indicating the probability of occurrence. After measuring one of the parameters, which is 1 in this example, PDF of this function should ideally reduce to a one dimensional

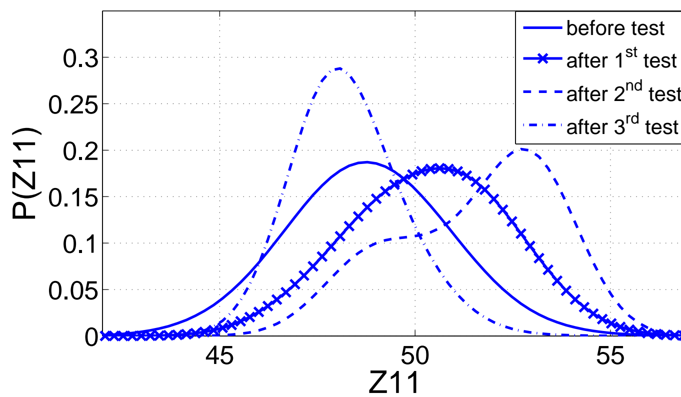
distribution defined in unmeasured specification parameter space. Due to the limitations of finite sample size, that is not practical. Hence, we use our penalization method to estimate the reduced dimensional JPDF employing the information local to the measurement value. Reducing window width in measurement space enables us to localize to the information in the vicinity of DUT. In this figure, we normalized PDFs with different coefficients in order to fit them in the same frame.

A more practical example of the update procedure can be demonstrated on practical data. Figure 32(b) shows how the JPDF is updated for one of the specification parameters of one experimental circuit (LNA) we employ in results section. Initially, the distribution of the JPDF in the specified dimension is wider and corresponds to the cumulative information of all device ensemble. However, as we keep updating the JPDF with measurement results, the PDF of specified parameter narrows. Figure 33 shows estimated PDFs and actual values of the same parameter for randomly selected 4 different device instances. This plot shows that the method successfully localizes to the correct region in the parameter space. Similarly, PDFs of other unmeasured specifications are estimated and only tests that have the potential to fail are executed.

This adaptive update procedure yields excellent results when most of the fails are due to marginal defects and collective statistical information can be relied upon. However, circuits may contain structural defects that break down the learned correlation information. Therefore, care must be taken before committing a final pass label on devices when the test set is not exhaustive.



(a)



(b)

Fig. 32. Joint probability distribution (JPDF) is represented using multi-dimensional kernels. (a) High dimensional distribution is marginalized after incorporating each measurement result, effectively shrinking the number of dimensions. This enables to narrow the possible region of the DUT in probability space to make more informed decisions about the DUT. (b) Estimated PDF of parameter Z11 is shown for several update steps for a single device.

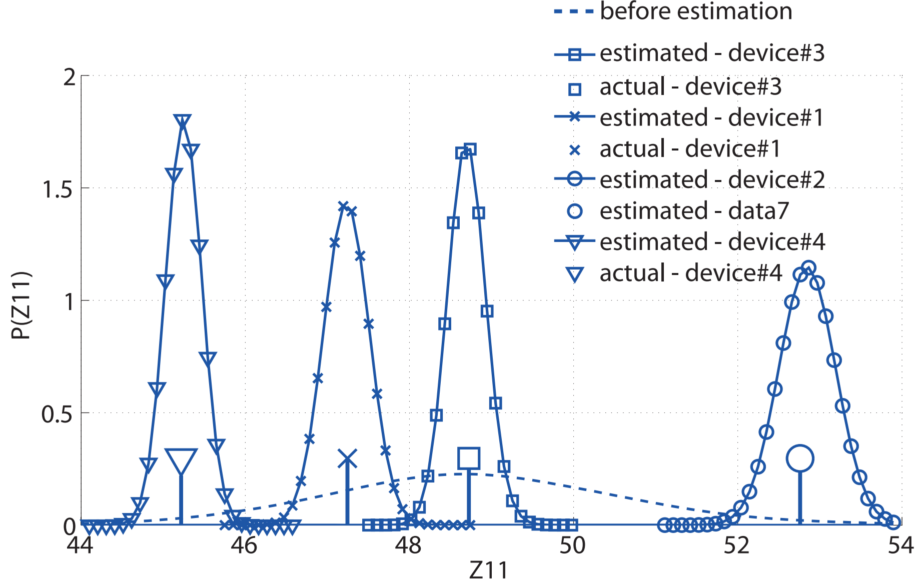


Fig. 33. Estimated and actual values of Z11 are shown after several update steps for 4 different device instances.

1.5. *Identifying Potentially Defective Devices*

Structural defects alter the circuit behavior in unpredictable ways and structurally defective devices do not conform to expected statistical distributions. Under this premise, the only certain way of ensuring that structurally defective devices that fail any specification are screened out is to apply the exhaustive test set to all the devices. However, this would result in unreasonable test cost as discussed earlier and can only be acceptable in certain application domains (e.g. automotive). The irregular behavior of structurally defective devices however, can be turned into an advantage and incorporated into the adaptive test flow. Since these devices do not display the expected characteristics, we can use this information to identify potentially defective devices and subject them to more testing even if they pass all measured specifications and are determined to pass the unmeasured ones. In order to enable this process, before finalizing a pass decision on a device, we conduct

additional levels of screening based on already measured specifications and decide whether further testing is necessary.

1.5.1. *Screening Step #1.* Even if the DUT satisfies the necessary conditions in order to be identified as a good device, two potential pitfalls exist: (a) the circuit may contain a structural defect, invalidating the correlation mechanisms among the specifications, (b) due to process shifts (new wafer or lot), the collective statistical information is no longer representative.

In the first screening step, the measurements obtained from conducted tests are used to gauge the deviation of the device from the collective mean value or its expected JPDF. In this step, we remove the assumption that the DUT should have similar statistics as the cumulative data in order to identify defective devices. Since we cannot use correlations between the specification parameters, we use cumulative statistics and not the updated JPDF in this step. Each measurement result is shifted to zero mean and normalized according to cumulative statistics given in equation (3.9) and then compared to a threshold value, Th_{S1}.

$$\frac{m_j - \mu_j}{\sigma_j} \quad > \quad Th_{S1} \tag{3.9}$$

If any of the measurement results exceeds that threshold value, the DUT is marked as potentially defective and passed to the second screening step. Screening is illustrated in Figure 34, where the gray region shows the PDF of a specification parameter, and *plus* signs represent where the devices in this particular example fall. In the figure, the first screening step classifies the three devices falling close to the cumulative mean as good, while the device falling beyond the threshold level is identified as a suspicious device.

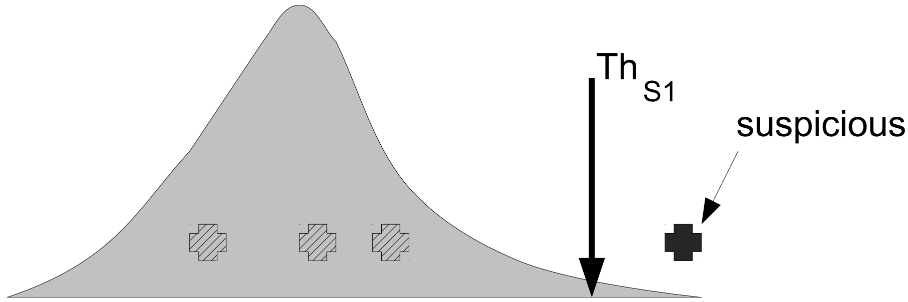


Fig. 34. The devices that fall beyond a pre-determined threshold level are deemed suspicious regardless of the specification limits. This enables to spot potential defects and reduce the chance of misclassification by conducting more tests to the DUT.

Note that this step uses already measured specification parameters; therefore, no additional test time cost is associated with this step. Flow diagram of the screening steps is illustrated in Figure 35.

1.5.2. *Screening Step #2.* If the DUT fails the first screening step, it is passed through the defect oriented second screening step to increase pass/fail confidence. In this step, we aim at pruning defective devices that do not correlate with the majority of the devices. If the DUT is a defective device, then correlation among the specification parameters that we employ for adaptive testing will no longer be valid, hence tests that are skipped using this correlation information will also be invalid. Therefore, in order to get more information about the DUT, we need to conduct some of the skipped tests. However, these tests must be selected such that identification should be achieved with minimum additional test overhead.

First we sort the initial test list using cumulative statistics such that tight specs are placed on the top of the list, where the tightness measure is $\frac{\mu_j}{\sigma_j}$. This operation does not depend on the input from DUT, so sorting the test list for tightness is performed once, before the adaptive testing phase starts. Hence, it does not contribute any computational

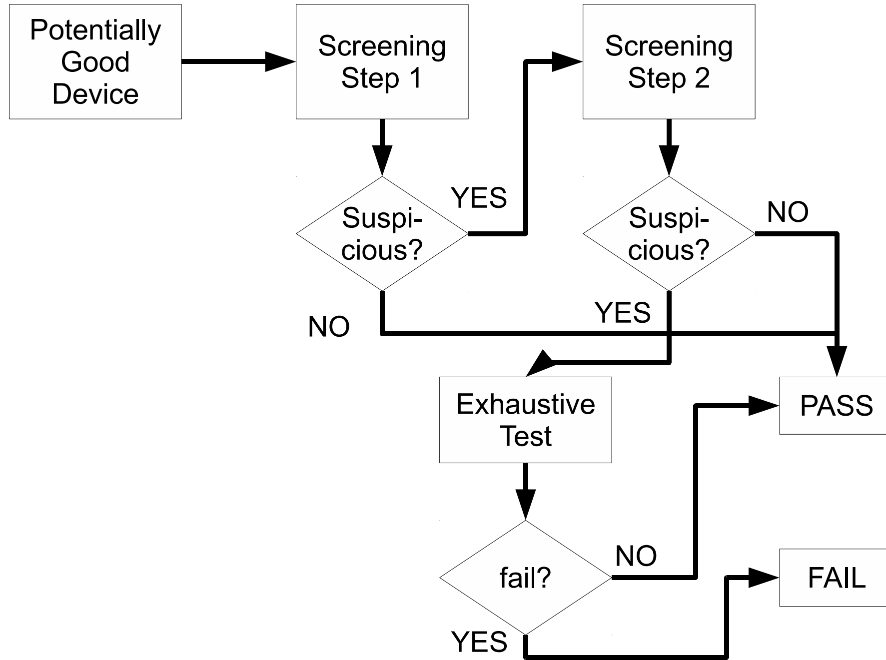


Fig. 35. Potentially good devices go through two sanity check steps to prevent defective escapes. DUT is analyzed statistically for suspicious behavior and tested further to minimize a risk of misclassification.

overhead in the adaptive phase. This sorted list is reduced by taking the first several tests and then re-sorted according to the same tightness criteria, however, using adaptive JPDF statistics this time. Finally, the first few tests in this list are conducted and irregularities of the new measurement results are checked using equation (3.9). However, a different threshold level (Th_{S2}) is used in this step. If the threshold is exceeded, all skipped tests are conducted exhaustively.

1.6. *Determination of The Initial Test List*

The adaptive method we propose follows a fixed test order throughout the algorithm, where the initial order is determined according to the statistics of the training sample set.

The main purpose of ordering the initial test list is to help the adaptive algorithm gain more information in fewer steps. As a result, a number of tests need to be applied first before the probabilities obtained from the kernel based estimator can be relied upon. Moreover, since defective devices and process shifts may alter some of the learned information, tests with more information content need to be applied before a decision can be made on the DUT. Thus an initial test list is necessary to apply the kernel based estimator. We use three criteria to select this initial list.

1.6.1. *Covering Test Condition.* As in most prior approaches, we use the fall-out data to select an optimum covering set. However, we update this information with each new data point encountered in the testing process. Since we force exhaustive testing on devices identified as potentially defective, we obtain information on the outcome of all tests. Hence regular updates to the test selection process are possible, where some tests that do not identify fall-out patterns anymore are dropped and replaced with new ones.

Note that since our objective is to reduce DPPM, we trade-off test time for test quality. Forcing use of a minimum number of tests puts a lower limit on test time, however, as we show in the results section, test quality is greatly improved.

1.6.2. *Marginality Condition.* The second condition we use for test ordering is the marginality of the specifications. We define marginality as the distance of specification limits from the specification mean in terms of standard deviation, as we show in Figure 36. Marginality measure is the scaled version of Cpk; Cpk measures the nearest distance in multiples of 3 standard deviations. If the distance (d_i/σ) of at least one of the specification limits is small enough, the specification has a non-negligible failing probability, which may have not been captured in cover based test selection step due to the limited size of the training set. Marginal specifications have large normalized variation. Due to the critical role

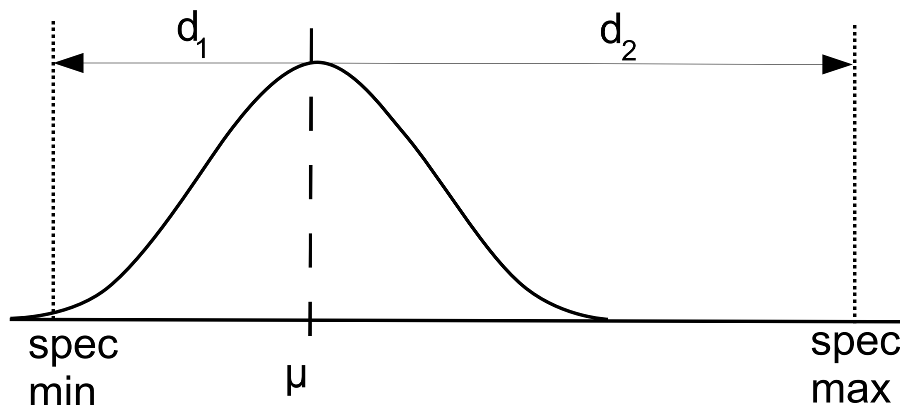


Fig. 36. Definition of marginality. Distance of the specification mean to the closest specification boundary is the measure of marginality. We normalize this distance with respect to standard deviation of the specification.

of marginal tests in lowering DPPM, we generate a marginal test list, where specification limits are closer than 3.5σ to the specification mean.

1.6.3. *Specification Tightness Condition.* The third condition we use to select the initial test list is the tightness of the specifications. We define specification parameter tightness as $\frac{\mu}{\sigma}$, where μ is mean and σ is standard deviation. We chose tightness as a test selection criterion since it reveals the most information for the adaptive test flow. Moreover, defective devices may have random specification violations that are neither marginal nor included in the covering test list. For those devices, we cannot use any predictive test selection algorithm. Instead, we approach this problem from the opposite angle, and choose first few of the tightest tests. The tightest tests are typically very unlikely to fail. However, note that this assumption is valid for fault free devices. Since, we aim at identifying defective devices, which behave radically different from the fault free devices, selecting tightest tests yield the most information. As an example, suppose that we have two non-marginal specifications

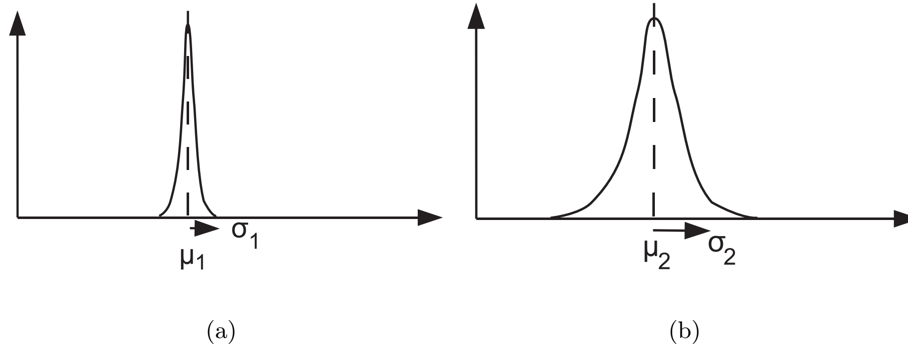


Fig. 37. Tightness of a specification is defined as $\frac{\mu}{\sigma}$. Therefore ,distribution in (a) is tighter than (b). Specifications that are tighter are considered to bear more information.

as shown in Figure 37, where the first specification is tighter than the second one. Under the assumption that defects generate unpredictable behavior, it would be more appealing to choose the first specification parameter to test since there is a larger probability for a defective device to fall further from the expected collective distribution.

1.7. *Characterization Data Compaction*

Characterization data obtained in training step is n vectors of specification parameters, where the size of each vector is equal to the number of specification parameters, M. Probability of failure computations during the adaptive test phase requires calculation of fail probabilities and updating of these probabilities. Hence, computational complexity of the method is proportional to the size of the characterization data. In order to mitigate the computational overhead, we develop a characterization data compaction method.

The basic principle of characterization data compaction step is illustrated in Figure 38 for a simplified compaction problem, where there are only two specifications. Horizontal axis and vertical axis show the location of the measurement result of the training samples for the first and the second specification parameter respectively. *plus* shapes represent sample

device instances on this scatter-plot. Initially there are 16 training samples in this particular example. However, we merge some of the instances and assign a weight which is equal to the number of merged instances. We define a radius, r, and merge the instances that fall in proximity of that distance. In the figure, shaded circles represent the r neighborhood of the samples. In this example instances falling in the same circle are merged together. Hence, 16 training samples are merged to yield a total of 5 merged instance in this example.

The merging algorithm starts with labeling all the instances in the training sample set as compact sample set, T_C, candidates. Then, the first instance is transferred from the candidate list to the list without compaction, since there is no instance yet to merge with. Then, we check whether the second candidate falls in the proximity of r with the first instance. If it is in the r proximity of the first instance, then the second instance is merged and the weight of the merged instance is incremented by one. Proximity of an instance is computed using equation (3.10), where $\|.\|$is L_2 norm, s_j, μ_j, and h_j are specification parameter value, mean and kernel width of the jth specification. If the computed value on the left of the equation is smaller than r then, the instance candidate is merged.

$$r \geq \sqrt{\frac{1}{n}\left\|\frac{s_j-\mu_j}{h_j}\right\|^2} \tag{3.10}$$

If the device instance is outside of the radius, then it is appended to the compact sample set and assigned a weight value of one.

1.8. *Training Set Update*

Training device instances are typically collected in production ramp-up phase using a small set of wafers, and this data is used to optimize the test list that is used in high volume production test. However, collected characterization data may become invalid due to time

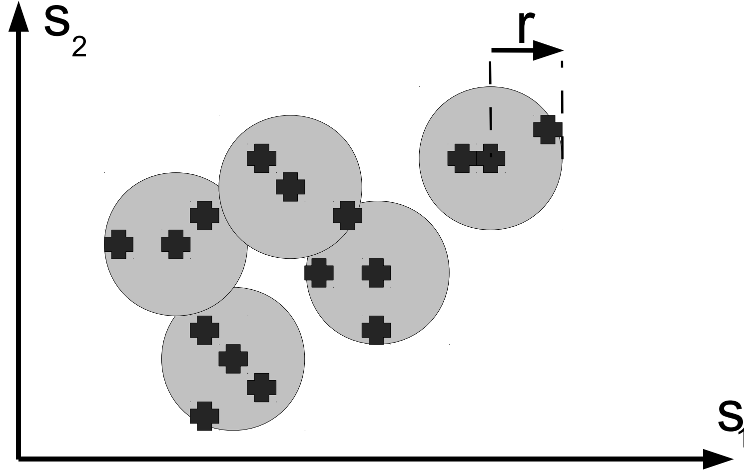


Fig. 38. Training samples that fall close than distance r are combined in order to compact the training sample set.

variant changes in the process state resulting in a potential increase in the defect escape level due to outdated information.

In order to detect a potential shift, we need a measure of difference to segregate two different process states. This is necessary to compare the information content of the available process state (using characterization data) and on-the-fly measurement data to decide whether the process has shifted or not. Moreover, we need an efficient re-learning strategy that will enable to update characterization data once it is deemed invalid. Main challenges of the re-learning consists of learning as fast as possible to update the characterization information in a short time, while maintaining the overhead at an affordable level.

We address characterization data invalidation issue in two steps. Firstly, we determine when the characterization data is outdated by continuously monitoring specification parameters. We use Kullback-Leibler distance (KL) to detect changes in the process state for each specification parameter separately. KL distance is a measure to calculate the relative

entropy of two distributions that shows how different two distributions are from each other. We use KL distance to measure the difference between the characterization data and a small sample set of current data to measure the rate of change in the process. Secondly, we update the characterization data set by appending new data until the KL distance drops within an acceptable range. We randomly select several devices at a certain rate for characterization and add them to the characterization data set. Devices are selected randomly to prevent skewing the original characterization set. The rate the devices are randomly selected is modulated with the number of specification parameters identified to be outdated using KL distance. This enables to adaptively adjust re-learning rate and prevents over-characterization.

1.9. *Process Shift Detection and Update*

The goal of this step is to achieve a high detection rate at a cost of minimum test overhead. Process shift can be detected by analyzing specification parameter measurements. The easiest way of detecting shifts is to analyze parameters with respect to time, i.e using a moving average filter to estimate mean and higher order moments. However, this method is not very sensitive and potential shifts may be masked by random variation. Another option would be to use auto-correlation of the parameters, but auto-correlation function is limited to capturing linear dependencies only.

In this work, we use a probability domain approach since we are mainly interested in keeping the JPDF of the process up to date and doing so enables to capture nonlinear dependencies relatively easily. We generate distributions of specification parameters using the most recently tested devices using a sliding window approach and compare them to the distributions obtained using characterization data set.

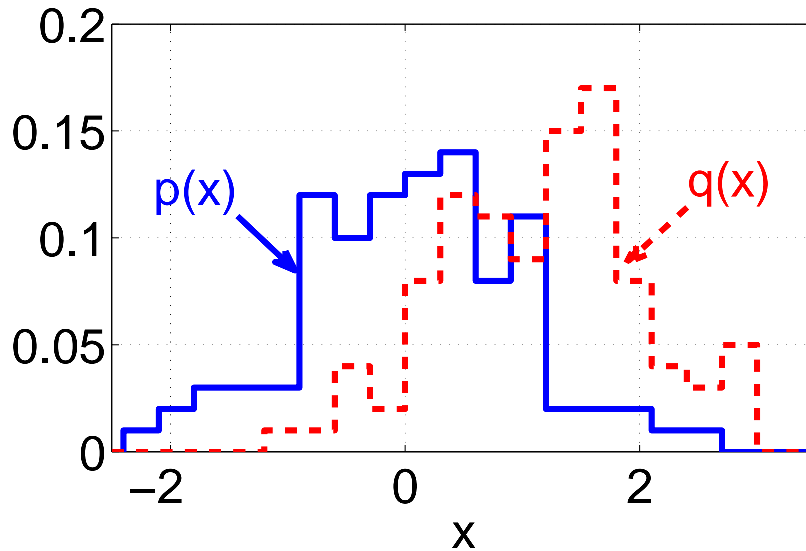


Fig. 39. KL-distance is used to measure the difference between two distributions. KL(p(x),p(x))=0, while KL(p(x),q(x))>0 if p(x) $\neq$ q(x)

There are a couple of parameters in this approach that have influence on the result and therefore deserve attention. The accuracy of the new distribution depends on two conflicting criteria that needs to be simultaneously met through appropriately adjusting the sliding window size. On one hand, a large number of samples (large window) is required to generate a high dimensional distribution. On the other hand, a small window is preferred to capture the most recent information about the process. First, we relaxed the large window size requirement to create a feasible solution by generating a distribution for each parameter separately instead of generating a multidimensional distribution. This greatly reduces the number of samples required for detection. Then, to further reduce computations, we used a histogram based distribution generation approach which generates dependable distributions using only 30-50 samples.

Once an up-to-date distribution is generated, it is compared to the distribution generated using the characterization data through KL distance given in (3.11).

$$KL(p \parallel q) = \sum_i p_i log_2(\frac{p_i}{q_i}) \quad (3.11)$$

Where, p_i and q_i are probability distribution p and q in i^{th} bin respectively. Figure 39 illustrates two distributions, p and q, and their corresponding KL distance calculated using (3.11). KL distance simply indicates how different distribution p is from distribution q. KL is zero if two distributions are the same or a positive number if they are different.

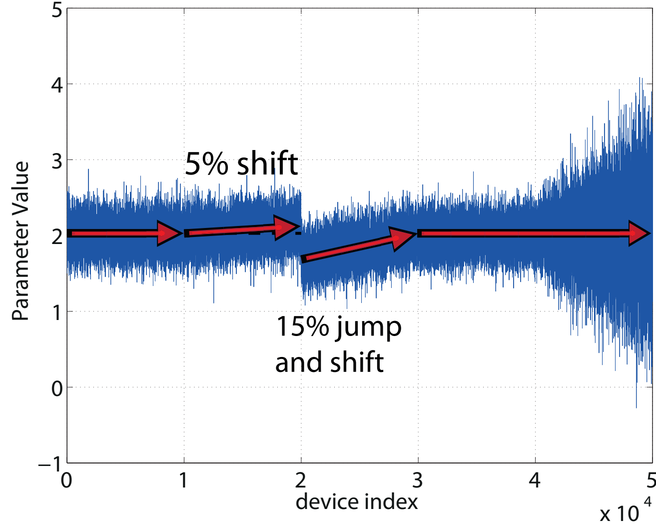
KL is very sensitive to changes in the parameter distributions and is able to cope with non-linear dependencies. Figure 40 illustrates how effective KL is in identifying changes in the process through a simple example. Statistical properties of a Gaussian stochastic process are altered in time an plotted in Figure 40(a). The process initially has a mean of 2 and variance of 1. The mean of the process is gradually increased up to 5% for the second 10k instances. After an abrupt jump, the mean approaches to the initial mean between 20k to 30k. Finally, variance is gradually increased between 40kth and 50kthsamples. The corresponding KL distance plot of the process is shown in Figure 40(b). The dashed curve shows the KL distance of $p_i(x)$ with respect to the initial distribution, $q(x) = p_0(x)$.

Distribution $p_i(x)$ is generated using the most recent 100 samples $(x_{i-99} \cdots x_i)$. Note that the dashed KL curve starts at 0 deviates from 0 significantly even for small shifts in the process in the second 10k samples. The jump and the increment in the variation can also be easily identified by the dashed KL curve. Note that the limited sample size introduces uncertainty in KL distance calculation, which is observed as noise in the KL plot. Therefore, we need to identify the region of uncertainty in order to differentiate process shifts from noise in calculating the KL distance. We generate auxiliary samples $(x_t^1 \backsim p_i(x))$ from distribution $p(x_i)$ and compute KL distance of the re-sampled distribution. This re-

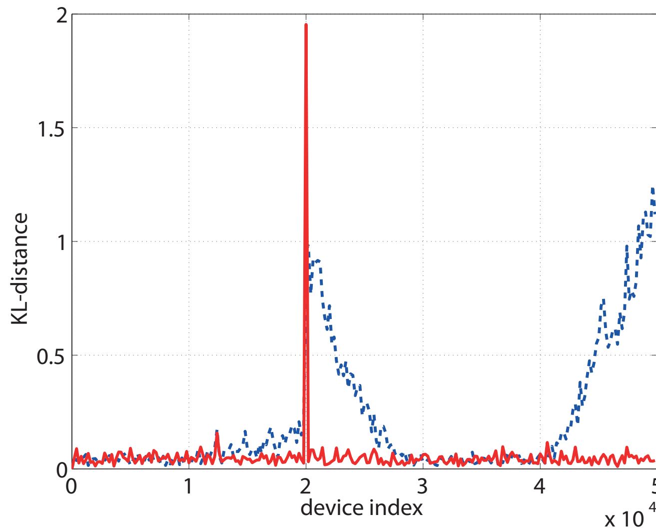
sampling approach enables to reveal the extent of uncertainty in the computed distance metric. We compute the KL distance of several re-sampled sets in order to estimate the first two moments of the shift-free statistics, μ_e and σ_e. Finally, we use equation $KL_{T,j} = \mu_{e,j} + 3\sigma_{e,j}$ as the threshold level to differentiate process shifts from the noise, where j is the index of the jth specification.

1.10. *Updating Characterization Set*

Characterization set needs to be updated to avoid degradation in test quality. One approach is to re-characterize the process periodically at wafer-to-wafer transitions or synchronize characterization to major interruptions in the process. This approach introduces significant test time penalty and cannot respond to changes within the specified characterization points. In order to improve reaction time and to keep characterization cost small, we employ an adaptive update scheme outlined in Figure 41. We subject randomly selected devices to the full test suite to continuously monitor the process. We define the rate that the devices are randomly selected as Random Test Rate (RTR). Randomly tested devices are used to compute KL distance of the process to the characterization set.



(a)



(b)

Fig. 40. (a) A non-stationary stochastic process and (b) KL distance of the process with respect to its initial probability distribution. The dashed curve reveals the changes in the process.

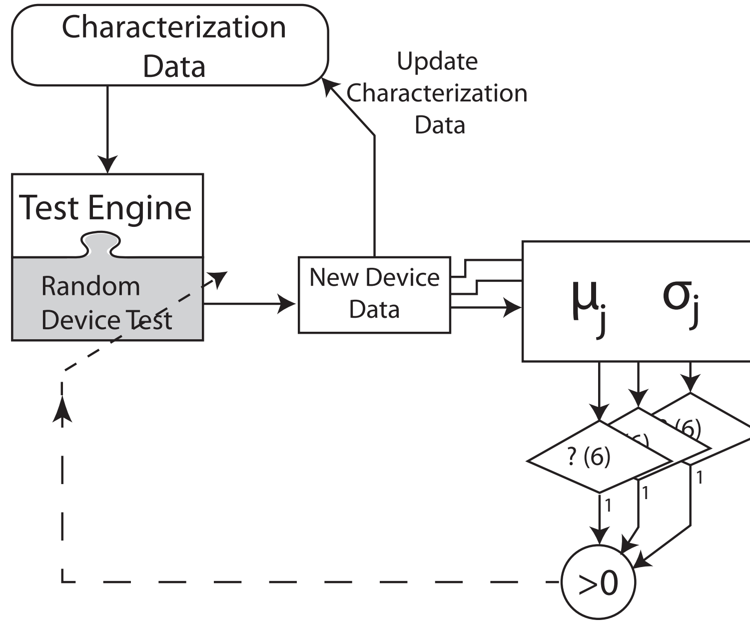


Fig. 41. Devices characteristics are analyzed for potential shifts in the process. Update is performed if one or more specification parameter statistics are outdated. The rate of re-learning is set with a RTR parameter to catchup with the changes in the process while not blowing the test time up.

There may be several sources of changes in the process leading to rapid or slow changes. Re-characterizing the process at every small change is wasteful leading to a large test time penalty. The penalty can be minimized if the characterization data is updated continuously at a rate proportional to the extent the characterization data is outdated. We obtain this information from the number of specifications identified to be outdated. For instance, if only one or two specifications are updated re-learning rate is relatively slow. However, if half of the specifications are outdated, the re-learning rate is much faster to capture up-to-date characteristics to prevent misclassifications. Re-learning rate is set by modulating the RTR parameter with the number of outdated specifications. This formulation establishes

a dynamic re-learning scheme that enables to re-learn only when necessary at a speed proportional to the outdated information.

$$RTR = \frac{\#\text{ of outdated specifications}}{\text{total }\#\text{ of specifications}} \tag{3.12}$$

1.11. *Computational Overhead and Pipe-lining*

There are several steps in our proposed technique that require computation power. However, only the steps that contribute to the on-line computation overhead are crucial. Hence, we analyze computational cost of on-line phase only. Offline steps are: training set compaction, initial test list generation, and test list generation for the second screening step.

On-line steps that contribute to computation overhead are: JPDF update, fail probability update/check and screening methods. JPDF update and probability update/check steps are optimized such that computationally expensive routines moved to offline phase. Only n_C kernel evaluations and two vector multiplications are performed in each adaptive test loop depicted in Figure 31, where n_C is the size of compact training sample set. Computational overhead of these steps and the first screening step are less than 2 ms. In addition to 2 ms, we have the computation requirement of the second screening phase, which is simply list re-ordering.

These computational time costs can be avoided through employing pipe-lined test methodology. In order to remove the processing time overhead out from the critical path, device testing and processing can be performed simultaneously. Figure 42 illustrates how testing and processing can be scheduled such that processing is moved out of the critical path. The leftmost column shows the time, while the other columns contains the schedule.

Time step 1	Test #1	
Time step 2	Test #2	Process #1
Time step 3	Test #3	Process #2
⋮	⋮	
Time step j	Test #j	Process #j-1 + sanity 1
Time step j+1	Sanity#2 test#1	
Time step j+2	Sanity#2 test#2	Process Sanity#2 test#1
Time step j+3	Sanity#2 test#3	Process Sanity#2 test#1

Fig. 42. Pipe-lined Time Schedule

The first step is conducted in the first time slot, and no processing is done since the result of the first test is not available until the end of its time slot. In the second time slot, the second test is executed and the result obtained from the first step is processed. The rest of the testing and processing steps are scheduled similarly, hence processing steps do not lie on the critical path.

Note that even though the first few tests are applied regardless of the outcome of the kernel based predictor, the precessing does not have to wait until all mandatory tests are exhausted.

1.12. *DPPM Estimation*

Adaptive test method identifies potentially good devices before conducting all tests using a stopping criterion ($pFail$), which is the failure probability of fault free devices. The proposed method identifies devices satisfying $pFail$ criterion as good devices and the result of this is a non-zero defective parts per million (DPPM). We can calculate DPPM using equation (3.13), where $\Pr(pass|bad)$ is the probability of a device being identified as

good although it is defective. The relation between $pFail$ and $DPPM$ is given in equation (3.14).

$$\begin{aligned} DPPM &= \Pr(pass|bad) \cdot 10^6 \quad (3.13) \\ DPPM &= pFail \cdot 10^6 \quad (3.14) \end{aligned}$$

Discussion given above is valid only if the DUT shares the statistics of the cumulative data. However, there are two mechanisms that alters the above formulation, which violates our assumptions. If the characterization data is not a good representative of the DUT in the statistical sense, then the constructed JPDF is not a good model, which will lead to mis-classifications. Misrepresentation typically occurs due to process shifts. Proper update of characterization data is necessary in order to avoid mis-classification errors related to mis-representation.

Another mechanism that degrades the test quality is random defects, which are caused due to unpredicted mechanisms. Devices affected from these mechanisms may have a completely different statistics, which makes them very hard to identify. Equation (3.15) shows the overall DPPM in presence of mis-representation and random defects. Last two terms in this equation constitute the DPPM floor.

$$\begin{aligned} DPPM &\cong pFail \cdot 10^6 + DPPM_{mis} \quad (3.15) \\ &+ DPPM_{random} \end{aligned}$$

Kernel based adaptive test methodology is responsible for controlling the first term of this equation. Other two terms require additional steps to be controlled. The term related

to mis-classification is addressed in the screening steps while the term related to random defects is addressed through test forcing.

1.13. *Results*

The proposed adaptive test flow is applied to three experimental circuits. The first circuit is a low noise amplifier (LNA) that we have designed. In the absence of production data, we rely on Monte-Carlo simulations to generate the device population. The second circuit is a diverse mixed-signal circuit with 42 specifications and many building blocks. We analyze roughly 89k sample devices of production data for the second circuit. This production data spans multiple wafers and lots and is a good representative set for this device. The third circuit is a small-scale analog device. We analyze 21k samples that also span across multiple wafers and lots for the third circuit.

We apply our method to all three circuits based on a small set of initial training data and use the rest of the samples to calculate test time and DPPM. We also compare our adaptive flow with static compaction methods published in the literature.

1.13.1. *LNA Results.* We employed a variable gain LNA shown in Figure 43 to evaluate the proposed method. In order to model process variation, parametric variation is injected to the circuit. Process variation injected to circuit parameters is shown in Table 7, where *DD* row shows die-to-die variation and *MM* row shows mismatch. 12 specification parameters are selected with 4 different gain setups, hence the total test list size is 48. Specification parameters are gain, bandwidth, center-frequency, input/output impedance, input/output matching, IIP3, 1dB compression point, power consumption, noise figure, and output offset.

Simulation results for 48 specification parameters are generated for 60k circuits. We choose training set size to be 2k and used the same training set size for the methods that

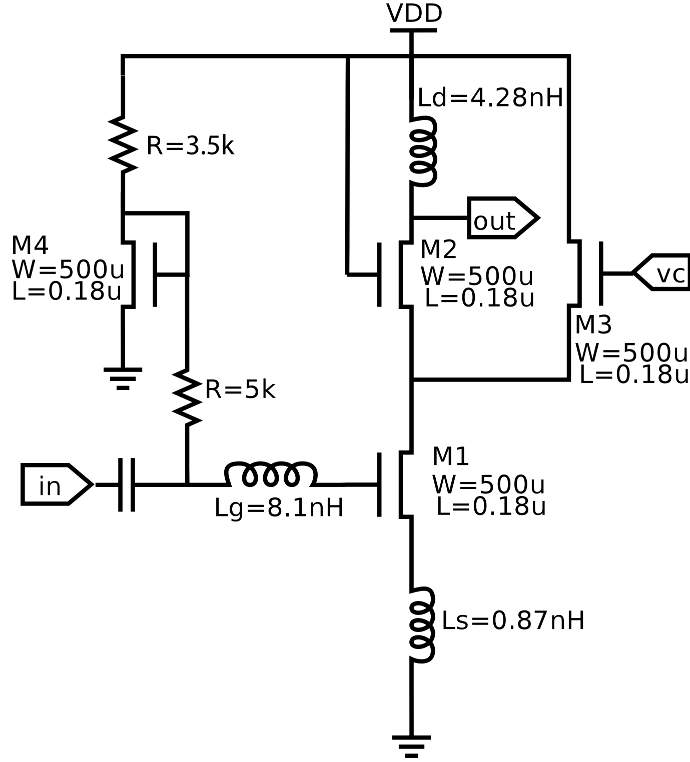


Fig. 43. Experimental circuit: LNA

	Width [3σ]	Length [3σ]	V_{th} [3σ]	L [3σ]	R [3σ]
DD	1 %	16 %	16 %	14 %	14 %
MM	0.06 %	2 %	2 %	2 %	2 %

Table 7. Process variation table

we compare with for fair comparison. The rest of the mutually exclusive 58k devices are used for verification. In order to avoid the dependence of test quality on the training sample set, test quality evaluation is performed several times and average test time and test quality is reported. We repeated test quality evaluation for mutually exclusive training sample sets, which is 30 for LNA case.

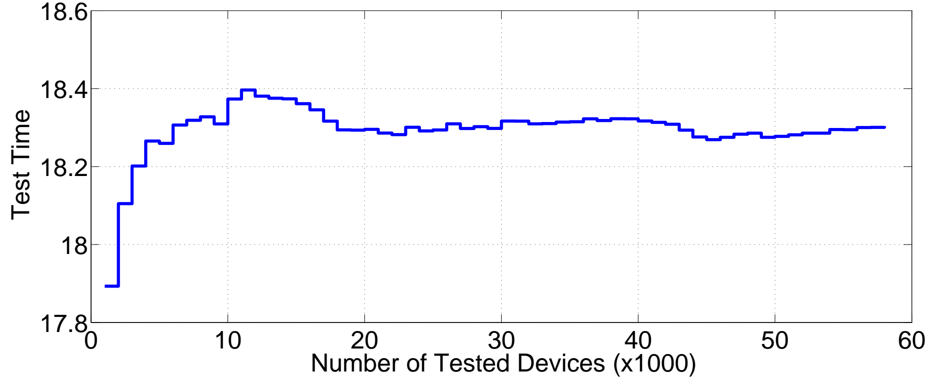


Fig. 44. Average test time during testing. Test time remains more or less constant despite of the adjustments in the test list and characterization data updates.

	DPPM	Time
Proposed	40	18.5
Cover Based	12.5k	18.4
ILP [Stratigopoulos *et al.*(2007)]	1k	26
Marginality Based [Chen and Orailoglu(2008)]	9.4k	4
Heuristic [Milor(1998)]	2k	24.3
static SVM [Biswas *et al.*(2005)]	11.6k	33

Table 8. Comparison of test compaction methods for LNA

Test quality evaluation results of our proposed method and several other methods are shown in Table 8. Result of each method is presented with a DPPM-time column pair. In order to show the capability of quality control, we tuned our method to get minimum possible DPPM level. Results show that DPPM of 40 is achieved for a test time of 18.5 tests, which is the best quality result. We compare results with a heuristic set-cover based algorithm for which we can adjust test time by forcing a certain amount of tests to be

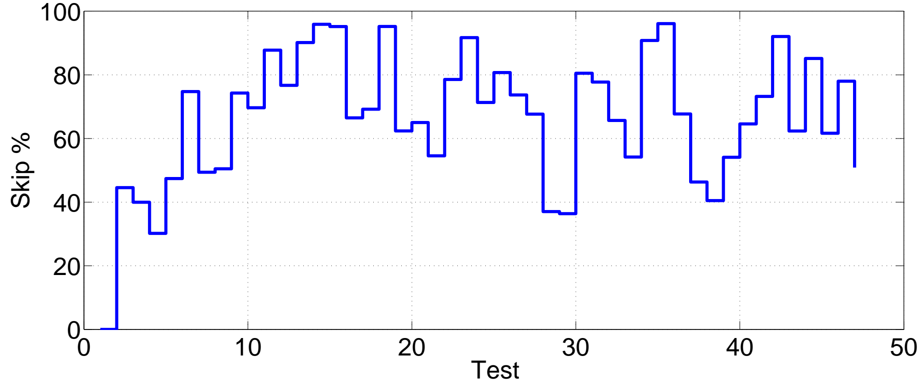


Fig. 45. Percentage of skipped tests.

conducted. Hence, test time of the heuristic method can be crudely matched and we can compare the test quality improvement for the same test time. Compared to cover-based method, test quality improvement is more than two orders of magnitude for the same test time.

DPPM and time metrics are generated for 30 runs with mutually exclusive training samples sets. We observed that the proposed method yields consistent results. In the next section we show that consistency is achieved on production data also. Figure 44 shows the average test time as the testing proceeds. The figure shows that the test time of the proposed methods is not sensitive to the process variation and it asymptotically reaches an equilibrium as testing progresses.

In this work, we aim at test quality improvement through treating each device according to its performance and proceed with testing by predicting its behavior. Hence, we drive the test mechanism according to each DUT and customize tests in order to achieve the best test-quality to test-time trade-off. Figure 45 shows that device specific test tailoring is a good method. The figure shows the percentage of test skips for LNA data for each test. The first few tests are not skipped, since enough information is not obtained for test

	DPPM	Time
Proposed	36	22
Cover Based	392	20
ILP [Stratigopoulos *et al.*(2007)]	800	5
Marginality Based [Chen and Orailoglu(2008)]	135	15.5
Heuristic [Milor(1998)]	590	14.34
static SVM [Biswas *et al.*(2005)]	735	17.4

Table 9. Comparison of test compaction methods for production data.

skipping. However, after the first few tests, the algorithm starts to skip tests at a very high rate yielding adaptive test list compaction.

1.13.2. *Production Data (Diverse Mixed-Signal Circuit).* We also apply the adaptive test flow to production data. We analyze 89K samples of the experimental circuit, which has 42 initial specifications. Once again, 2k samples are selected as training set and the rest of the devices are used to compute test time and DPPM. The process is repeated with multiple distinctive training sets, although the training sets are always drawn in a contiguous manner.

We first show the effect of characterization set updating algorithm. Figure 46 illustrates the KL distance of a randomly selected parameter using the proposed updating scheme (solid curve) and a non-updated scheme (dashed curve). The distance of the updated scheme remains below the non-updated scheme and we see a consistent reduction trend. Relatively small excursion in the solid curve suggest that continuous updating scheme works and only incremental characterization is required.

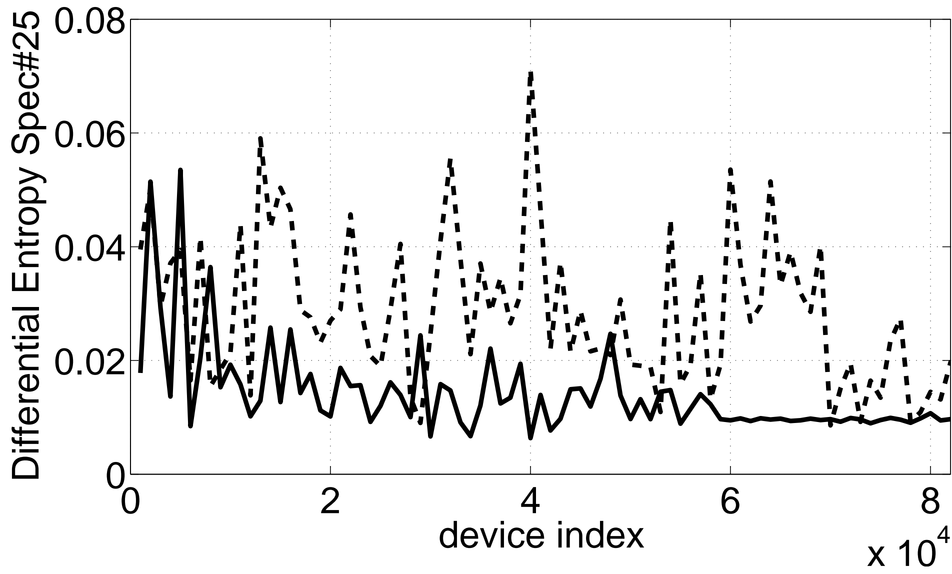


Fig. 46. KL-distance of updated (solid) and non-updated (dashed) curve shows the consistency of the characterization data converging to the up-to-date statistics.

Table 9 shows the test time and DPPM levels for our method and for static compaction methods published in the literature. The table shows that even though ILP formulation provides the best test time, the DPPM level is the largest. If test time is the only concern, such a static compaction technique may suffice. The table also indicates DPPM values compared to our adaptive flow. The proposed method achieves a significant DPPM reduction compared to the other methods. Compared to the cover based method, which we implemented to represent the traditional test compaction methodology in industry and modified to be able adjust for a desired test time, 10 fold improvement in test quality is observed for the same test time.

1.13.3. *Production Data (Small-Scale Analog Circuit).* We analyze production data of 21K samples of this experimental circuit which has 21 initial specifications. Similar to the

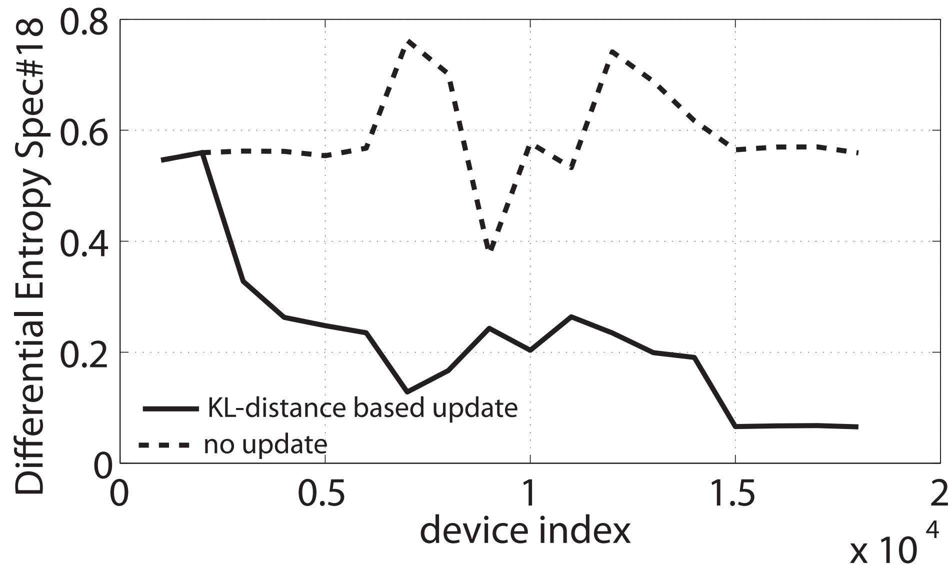


Fig. 47. Non-updated scheme suffers from invalid characterization data, therefore, is not dependable. However, the proposed updating scheme keeps characterization data up-to-date.

setup for the previous two experimental circuits, 2k samples are selected as training set while the rest of the devices are used to compute DPPM.

Figure 47 shows KL-distance of updated (solid) and non-updated (dashed) schemes for a randomly selected specification parameter. This particular parameter experiences an early shift as KL distance starts at a high value and remains more or less at the same level in the non-updated case. However, the solid curve approaches to zero, indicating that error in representing process statistics is consistently reduced.

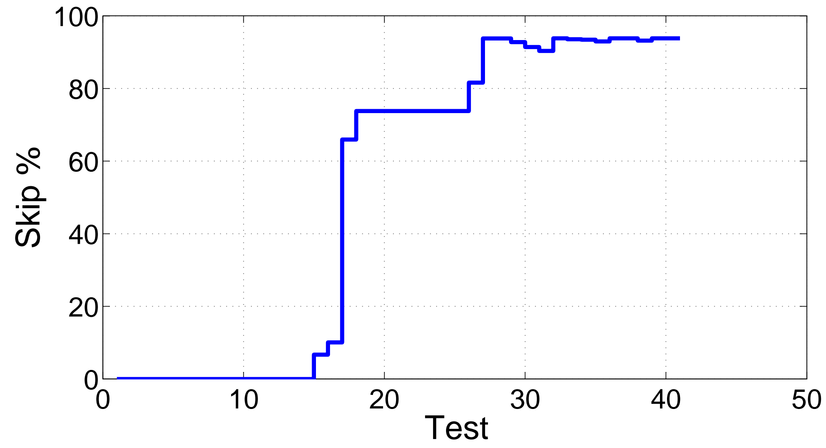
Results for the second production data are shown in Table 10. ILP and marginality based methods achieve low test times, but DPPM is significantly high. Therefore we reach to the same conclusion for both production data; static test compaction may achieve a good test time reduction, however, test quality is typically poor.

	DPPM	Time
Proposed	130	10
Cover Based	1.4k	7
ILP [Stratigopoulos *et al.*(2007)]	1k	3.5
Marginality Based [Chen and Orailoglu(2008)]	1.92k	3
Heuristic [Milor(1998)]	150	6.56
static SVM [Biswas *et al.*(2005)]	664	17.2

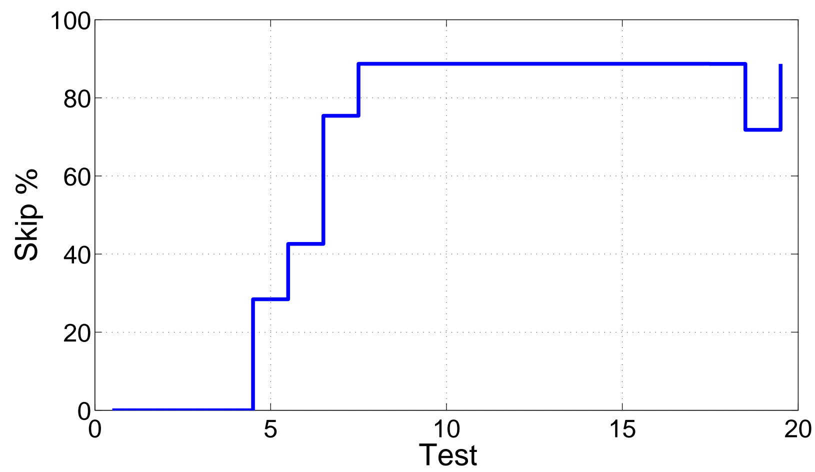
Table 10. Comparison of test compaction methods for production data (analog signal circuit)

Note that marginality based method achieves a good result for mixed-signal circuit data set, while Heuristic method achieves a good DPPM result for small-scale analog circuit data set. This suggests that these methods can yield good results but their performance varies depending on the circuit and statistical properties of the process; therefore, they may also result in catastrophic results. In that sense our proposed method is robust and shows consistent performance improvements.

Adaptability of the proposed method is based on our claim that devices are coming from a diverse set and per-device based optimization achieves a better test time to test quality trade-off. We support our claim with test skip histograms as we did for the LNA circuit. Test skip histograms for mixed-signal data and small-scale analog data is shown in Figure 48(a) and Figure 48(b) respectively. None of the tests are skipped at the beginning due to insufficient confidence in passing the devices; however, once the desired confidence



(a)



(b)

Fig. 48. Skip histogram for (a) mixed-signal industrial circuit (b) analog industrial circuit

is achieved most of the remaining tests have an almost equal skip rate, indicating that each test is equally likely to be measured based on the characteristics of the DUT.

1.14. *Summary*

Existing test compaction methodologies are not built with test quality control mechanism, hence they yield unacceptable levels of test quality degradation. In this work, we investigate a per-device adapting test list compaction methodology, which utilizes on-line measurements to tailor an optimized test list. Moreover, we continuously monitor the process for potential shifts and update when necessary. Thus, characterization data remains valid throughout test lifetime. This strategy enables us to devote test resources optimally to each individual DUT in the sense that marginal devices are devoted more test resources while devices falling close to nominal are passed with less testing. Each DUT may be subjected to unique sequence depending on its behavior. Hence, a good test quality-test time trade-off is achieved. Moreover, the inherent quality control mechanism of the proposed method enables us to achieve a very low DPPM level. In addition to using probability distributions of specification parameters to decide when to stop testing, we also provide additional defect screening mechanism, which increases the confidence of pass decision. Experimental results of simulation data of an LNA circuit and production data of a mixed-signal circuit and a small-scale analog circuit showed that our proposed method consistently achieves a low DPPM level, which is at least one order of magnitude lower compared to traditional static compaction methods.

2. Adaptive Multi-site Test for Analog/Mixed-signal Circuits

Increasing integration packs more functionality in a single chip necessitating the testing of even more specification parameters. However, there is a tendency to keep test time budged constrained, which leads test engineers to seek more efficient test strategies. Statistical test compaction methods offer generic and circuit independent means of achieving efficient testing. Adaptive test methodologies have been shown to achieve better test quality versus test time trade-off compared to non-adaptive methods. In this work, we propose a new adaptive test approach geared for multi-site applications to achieve a significantly better test time/test quality trade-off. We employ an innovative compound-device approach that enables us to exploit device-to-device correlations. Moreover, we use neighbor device statistics for efficient defect screening. We show that despite the constraints imposed by multi-site testing, we successfully reap the benefits of adaptive testing in a multi-site environment.

2.1. *Adaptive Test in Multi-Site Environment*

We base our method on the adaptive test flow presented in [Yilmaz and Ozev(2010)], illustrated in Figure 49(a). In the production ramp-up phase, the full test suite is applied to a set of devices for characterization. This information is used to capture specification-to-specification dependencies and to generate an initial test list. In the high volume manufacturing phase, each device is tested using the flow shown in the figure. The core of the flow, shown in Figure 49(b), is an estimation engine that uses measurement results of the executed tests to predict failure probabilities of non-measured tests. This estimation engine effectively reduces the collective statistical distribution of an unmeasured parameter to that of its conditional distribution based on measurements already conducted on the

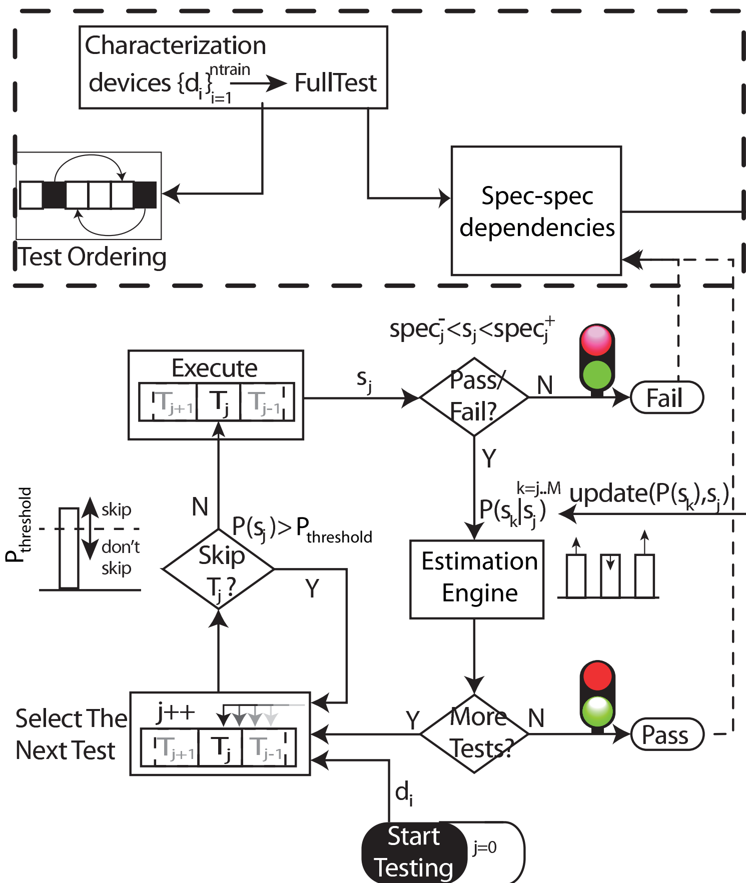
device. The conditional probability distribution is guaranteed to be tighter if there is a statical correlation between the measured and unmeasured parameters [Papoulis and Pillai(2001)]. This can be explained in the following manner. Suppose that X represents an unmeasured parameter and Y represents a measured parameter. The conditional variance of X based on an observation Y is given by [Papoulis and Pillai(2001)]:

$$var(X) = E[var(X|Y)] + var(E[X|Y]) \tag{3.16}$$

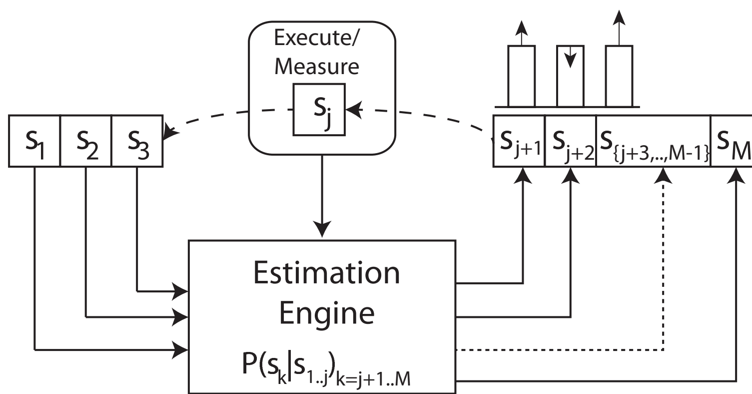
where, $E[.]$ is expected value operator and $var(.)$ is variance operator. Since both terms on the right are non-negative, equation (3.16) implies that $var(X) \geq var(E[X|Y])$. Thus, the conditional variance of X is always smaller than the variance of X unless X and Y are totally uncorrelated.

Since most parameters of a given DUT or even parameters of neighboring DUTs are structurally or spatially correlated, information obtained from each measurement is guaranteed to help in localizing an unmeasured parameter. This enables us to evaluate the pass/fail likelihood of the device under test (DUT) and provides a way to differentiate marginal devices that have the highest failing potential. Marginal devices (devices that have non-negligible failing probability) are assigned more tests to reduce the chances of misclassification, while non-marginal devices are passed with less testing to reduce test time.

This adaptive test approach exploits specification-to-specification correlations to reduce test time by eliminating unnecessary tests. However, there is typically a lower limit on the rate of test compaction that can be achieved with statistical methods. This limit depends on the complexity of the circuit and spec-spec correlations. In order to achieve lower average test time, parallel (multi-site) test approach is used. This approach scales average test



(a)



(b)

Fig. 49. (a) Adaptive test elimination flow and (b) Estimation engine.

	test#1..18	test#19	test#20	test#21	test#22	test#23	test#24	test#25	test#26
site#1		233	261	262	264				
site#2		228	233	261	262	264			
site#3		203	204	227	228	233	261	262	264
site#4	...*	233	261	262	264				
site#5		262	264						
site#6		264							
site#7		264							
site#8		262	264						

Fig. 50. Bulk of the devices require a relatively small number of test, while marginal devices typically require longer test times. Overall test time increases since all sites must wait until the device with the longest time is tested. * The first 18 tests are not show due to the limited space.

time per device linearly with the number of sites used for testing. As mentioned in the introduction however, a constraint of multi-site testing is that all devices must finish their testing before each DUT can be removed from the tester. Thus, even if one site contains a marginal DUT that requires more test time, all sites experience the same longer test time, effectively increasing the per-device test time before scaling. We illustrate this problem in Figure 50 by showing the executed tests for an 8-site test configuration using a production data set.

In this example, the average test time for 8 devices is less than 22 tests, while the device at site#3 requires 26 tests. Therefore, all sites experience the longest test time, which is the test time of site#3. Thus, instead of the average 21.4 test time over the 8 DUTs in this case the per-device test time is increased to 26 and the advantages of adaptive test diminish. Despite its constraints, multi-site approach also provides new means of improvement. Figure

53 shows a possible multi-site device testing configuration. In multi-site test, simultaneously tested devices typically originate from the same wafer and thus are spatially correlated in terms of their process parameters. This spatial correlation makes the measurement results of the device parameters highly predictable once their neighbors are tested. This approach fits very well in the adaptive test methodology. Statistical correlations among the devices that are tested simultaneously can be used to eliminate redundant tests.

In order to fully exploit the spatial correlations among the sites, we propose two techniques. First, we use a compound-device based approach for statistical modeling such that the prediction of one parameter includes the information that originates from neighboring sites. Second, reminiscent of variance reduction techniques from the digital domain [Daasch *et al.*(2000), Daasch *et al.*(2001), Madge *et al.*(2002b), Daasch *et al.*(2004), Madge *et al.*(2002a)], we use the neighborhood statistics to screen for defects, which helps in relaxing the initial test selection conditions.

2.2. *Compound Device Approach*

In order to extend adaptive test approach to multi-site test and to benefit from correlation between neighbor devices, we propose a compound device approach. Instead of treating each device individually, we combine all devices that are tested simultaneously into one compound device. L devices with M specification parameters are represented in the statistical estimation framework as 1 device with $L \cdot M$ specifications. Hence, the multi-site test problem is converted to testing of a single compound device. The only difference is that the compound device has L times more specification parameters and each test execution returns L test results. This compound device approach is illustrated in Figure 51. Once devices are converted to compound form, they are tested using the same flow

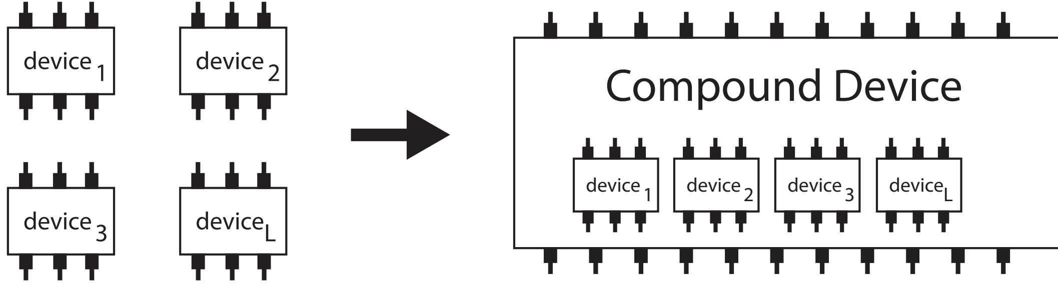


Fig. 51. Compound device. Simultaneously tested devices are treated as a single big device that has specification parameters. Where, L is the number of sites and M is the number of specifications per device.

that is used for individual devices, given in Figure 49. This approach enables us to model device-to-device correlations as well as specification-specification correlations and fits well in the test flow.

To show that device-to-device correlations are useful in estimating neighbor device characteristics, we present a parameter estimation example using a set production data. In this example, three consecutively tested devices are selected. We used the information on two tested parameters of two tested devices to predict the untested parameter of a third device. Figure 52 demonstrates prediction of device parameters using neighbor device measurements only. We would like to demonstrate that parameter#1 of a device can be estimated using measurements from two neighbor devices. Estimation is performed using maximum-likelihood (ML) criteria on the estimated distributions. The solid curve in the figure shows the distribution of parameter#1 based on cumulative information. Note that this distribution captures the common behavior of all devices in the representative characterization set. We would like to estimate the location of a particular device (device #1) by using measurement information from device#2 and device#3. Measurements from two devices are

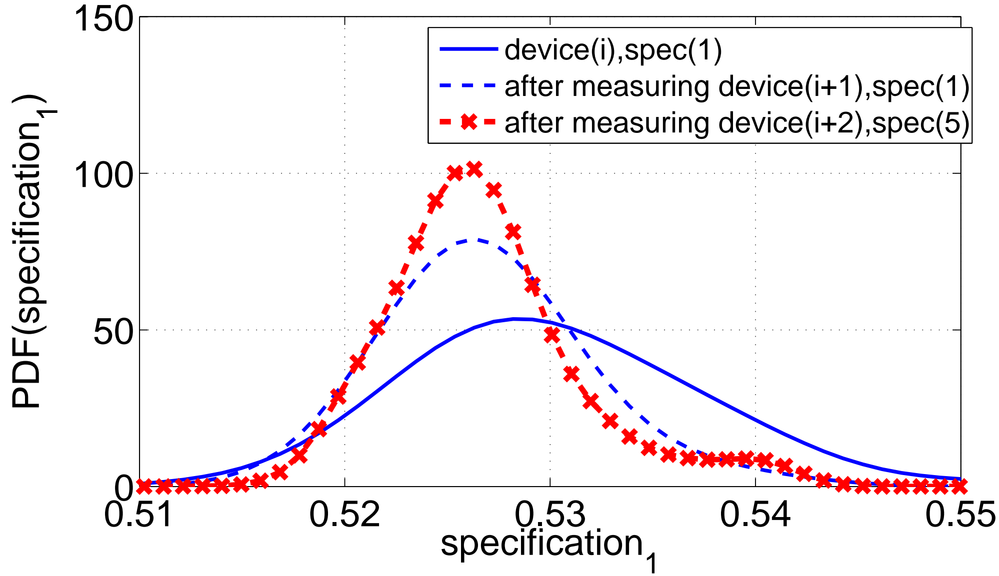


Fig. 52. Estimation of a device parameter using neighbor devices measurements only.

incorporated in the update procedure [Yilmaz and Ozev(2010)] and posterior distributions are plotted with a dashed and a dotted curve. We note that the initial distribution moves around the parameter being estimated and becomes narrower. In this demonstrative example, the ML estimate of the parameter approaches to the actual value using only neighbor measurements. Notice that the estimated distribution becomes narrower thanks to device-to-device correlation of neighbor devices. While the resulting posterior probability region is still quite wide and further measurements would be necessary before device#1 can be passed without measuring spec#1, this example clearly illustrates the spatial correlation of neighbor devices which can be use to make informed decisions.

2.3. *Determination of The Initial Test List*

In adaptive test, tests are scheduled such that an initial test order is not changed. A common, fixed test list is used and per-device adaptation is achieved by eliminating

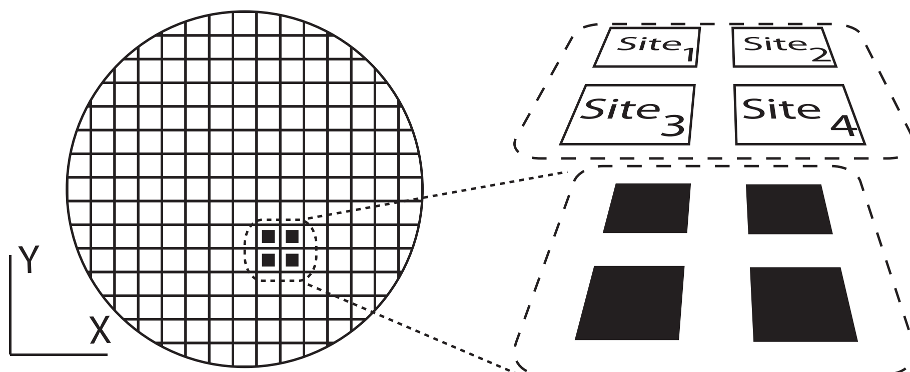


Fig. 53. Multiple devices are tested in parallel to increase the throughput. Devices being tested in parallel are typically neighbors on the wafer and are highly correlated

unnecessary tests. Thus, since the initial test list order is used for all devices, it should capture the collective behavior of all devices. In this work, we determine the initial order based on the incremental failing device coverage rate of collective data. Tests that have highest coverage rates are placed on the top of the list. This ordering scheme enables us to execute tests that have relatively higher potential to fail up front.

Although this ordering method yields a reasonably good initial order, using the same test order in all sites might not be the most efficient way of using tester resources. As the example in Figure 52 illustrated, distinct parameters of neighboring devices may yield good estimation opportunities using spatial correlations. Measurements of the same parameter are correlated in all sites. Therefore, instead of applying the same test, measuring relatively uncorrelated parameters might yield more information in less time. Thus, instead of using the same test order for all sites, we scramble the order of tests in blocks of L, where L is the number of sites. We circularly rotate the initial test to achieve a different test order for each site. However, we rotate the initial test list in groups of L, as shown in Figure 54, to ensure tests that have high coverage rate remain at the top of the list.

	site#1	site#2	site#3	site#4
test slot#1	1	4	3	2
test slot#2	2	1	4	3
test slot#3	3	2	1	4
test slot#4	4	3	2	1
test slot#5	5	8	7	6
test slot#6	6	5	8	7
test slot#7	7	6	5	8
test slot#8	8	7	6	5

Fig. 54. Initial test list is generated according to the coverage rate. A different order is assigned to each site by circularly rotating the initial test list in groups of , where is the number of sites.

2.4. *Neighbor Statistics Based Defect Screening*

Generally, test compaction based on statistical learning relies on correlations among specifications and works well for the bulk of the devices. However, there is a small number defective devices that do not conform to the collective statistics. Defects alter circuit structure such that defective devices are generally statistically distinct from defect-free devices. As such, relying entirely on statistical estimation for testing results in poor quality. In our prior work [Yilmaz and Ozev(2010)], we have addressed this problem by incorporating additional tests that do not provide more information about the bulk of the devices but have the potential of identifying statistical outliers. In multi-site testing, one advantage that we have is on-the-fly access to parameters of multiple devices which are expected to originate

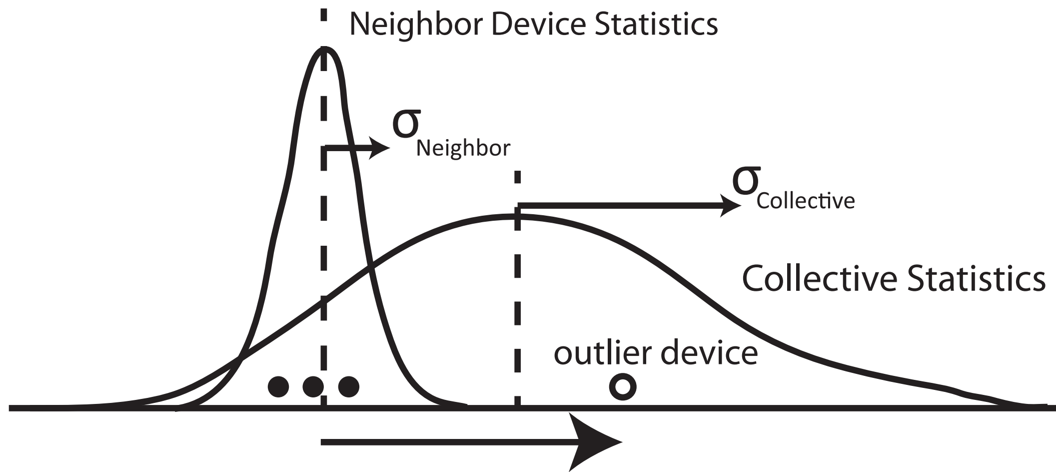


Fig. 55. Using neighbor device statistics enable us to use a narrower defect filtering window due to device-device correlations.

from very similar statistical distributions compared to the more representative characterization data. We thus use this multi-site information as a defect-screening mechanism so as to relax the requirement that we impose on the original test list.

We subject the measured parameters of tested devices that have not failed a test and have been identified as potentially good devices to a screening step. In this step, we determine whether a potentially good device falls outside of a $k\sigma$ window in the parameter space, where k is the screening window parameter. Devices that fall outside this window are exhaustively tested as opposed to being rejected. Thus, there is no yield loss. The mean and the standard deviation are determined based on the measurements of neighboring devices. This simple form of screening step is reminiscent of variance reduction techniques used in digital parametric testing [Daasch *et al.*(2000), Daasch *et al.*(2001), Madge *et al.*(2002b), Daasch *et al.*(2004), Madge *et al.*(2002a)]. However, we use multiple specifications for this purpose. Note that parameter k affects the number of the exhaustively tested devices and it is selected depending on the amount of increased test time that can be afforded.

We incorporate device-to-device correlations of neighbor devices to achieve a very effective filter. Figure 55 illustrates distributions generated using neighbor device information and collective device information. Collective distribution is wider, since it shows the common statistics of a large number of devices. However, neighbor device set distribution is much narrower due to high correlation between neighbor devices. Thus, using more descriptive neighbor statistics for defect filtering achieves superior defect screening performance. In Figure 55, we conceptually show the advantage of using neighbor statistics in screening. Filled circles show where the most of the devices fall, while the hollow circle represents an outlier device that is significantly different from the neighbor devices. However, if screened with collective statistics, the same device falls almost in the middle of the parameter space and therefore cannot be identified as potentially defective.

2.5. *Adapting to Process Shift and Computational Overhead*

The statistical decision making mechanism relies on the accuracy of the specification-specification dependencies. Although a reasonably accurate representation can be captured in the characterization phase, process statistics are subject to change over time. Thus, it is necessary to adapt to process shifts and update the correlation information to guarantee proper operation. One approach to do so would be periodic re-characterization.

We don't adopt such method due to excessive characterization time overhead. Instead, we use measurement information obtained from the screening step where a portion of the devices are exhaustively tested. We use this already existing information to update the dependency information that is used by the estimation engine. Nominally, only a small fraction of the devices should be exhaustively tested as the screening window size is selected to do so. However, an increase in the number screened devices indicates a potential shift

in the process. We use some of the screened devices that do not fail any specification test to update the characterization data.

Another concern in the proposed method is the computational overhead. We pipeline test execution and processing as proposed in [Yilmaz and Ozev(2010)] to take computations out of the critical path. The execution time of the statistical processing step is less than 1ms on a 2.6GHz computer, which is a fraction of the time that takes to conduct most tests. Therefore, test execution and test elimination processes are performed simultaneously without affecting each other.

2.6. *Results*

We use 2 sets of production data to evaluate the performance of our proposed method. The data consists of measurement results of the tests in the full test suite. The first data set has 42 specification parameters for a large scale mixed-signal circuit of ~89k devices. The second data set is of a large scale circuit with 264 specifications and ~900k devices. Throughout the analysis, we use disjoint training sets of 2000 devices for characterization and we use the rest of the devices of the same data set for verification. Performance results reported in this section are the average of 10 runs with disjoint characterization data sets. Reported test times are in terms of the average number of applied tests, while DPPM level is extrapolated from the estimated defect escape level to reflect the escape level per million devices.

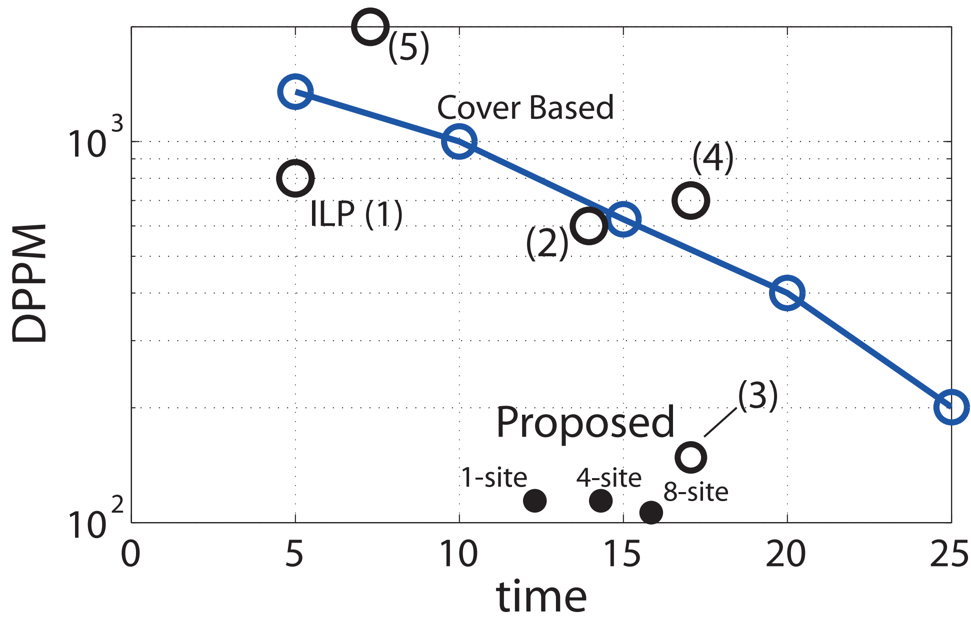
Table 11 shows the performance results on the first data set for 1, 2, 4, and 8-site test configurations. Approximately same DPPM level is obtained for all test configurations and almost linear scaling is achieved in test time. Slight deviation from linear time scaling can be attributed to the constraints imposed by multi-site testing.

Data-Set#1	1-site	2-site	4-site	8-site
Time	12.1	13.9/2	14.5/4	15.4/8
DPPM	105	90	112	86

Table 11. Test time and DPPM results for 4 different multi-site configurations. Almost linear time scaling is achieved with respect to the number of parallel sites.

We compare our proposed method with the previous work in Figure 56 on a DPPM versus test-time plane. This plot shows the performance of the previous work pictorially and enables us to understand in which DPPM vs test-time trade-off region the methods fall. Most of the previously proposed test compaction methods lack a quality control mechanism; they occupy a single point on the plot. In order to show the trade-off between DPPM and test time, we plot the results of the cover based method, in which the test list is generated by selecting highest incremental covering tests using a greedy algorithm. DPPM performance of the cover based method is plotted in Figure 56 using several different size test lists. Since our proposed method is geared towards achieving a low DPPM level, our method appears in the lower part of the trade-off plot.

Results for the second data set is shown in Table 12. Similar to the results obtained for the first data set, a linear scaling is observed in terms of time while DPPM level remains constant. Graphical comparison of our method is shown in Figure 57. Cover based method again serves as a reference line in the trade-off plot. Our proposed method is significantly below the reference line and achieves a very low DPPM level.



(1) [Stratigopoulos *et al.*(2007)]

(2) [Milor(1998)]

(3) [Chen and Orailoglu(2008)]

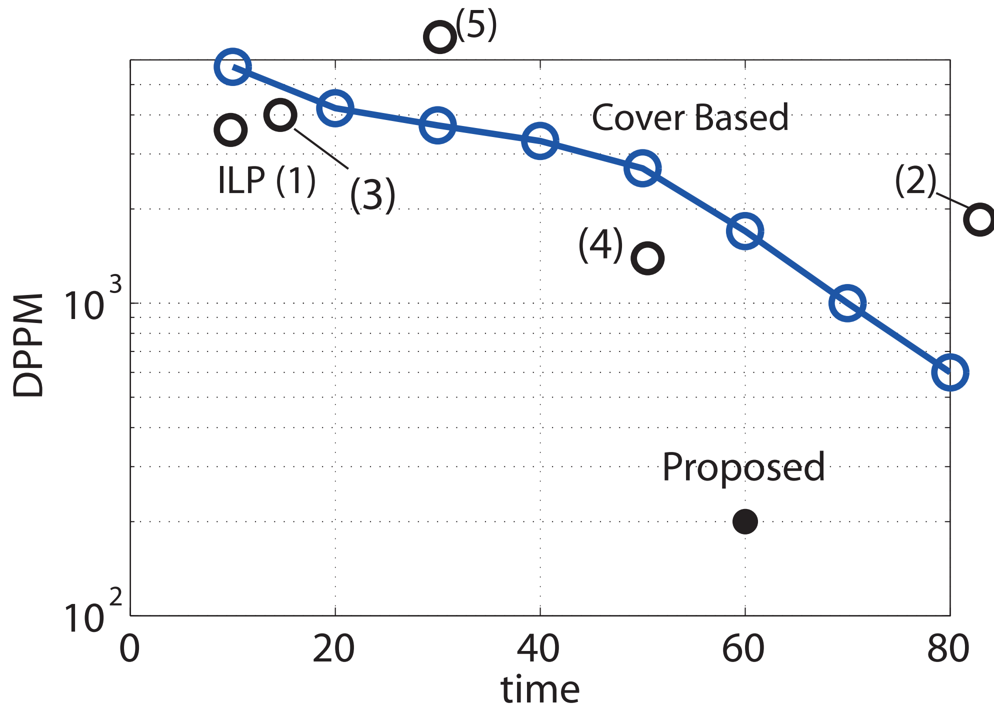
(4) [Biswas *et al.*(2005)]

(5) [Benner and Boroffice(2001)]

Fig. 56. The proposed method and previous work is compared. The curve of cover based method serves as a guide to DPPM vs time trade-off. Achieving lower DPPM requires longer test time and visa verse. * Test time of multi-site test schemes are scaled up (multiplied) by the number of sites for fair comparison.

2.7. *Summary*

Adaptive test elimination framework enables us to maintain a very low DPPM level and efficient utilization of test resources to reduce test time by tailoring the test flow to the statistical characteristics of the DUTs. To achieve significantly more reduction in test time,



(1) [Stratigopoulos *et al.*(2007)]

(2) [Milor(1998)]

(3) [Chen and Orailoglu(2008)]

(4) [Biswas *et al.*(2005)]

(5) [Benner and Boroffice(2001)]

Fig. 57. Cover based method can be used as a reference for performance comparison. This figure shows the performance of previous work on DPPM vs time trade-off plane. * Test time of multi-site test schemes are scaled up (multiplied) by the number of sites for fair comparison.

Time	1-site	2-site	4-site	8-site
Time	62	61/2	62/4	63/8
DPPM	206	220	180	185

Table 12. Test time and DPPM results for 1, 2, 4, and 8-site configurations for data set-2. Almost linear time scaling is achieved with respect to the number of parallel sites.

multi-site testing is desirable. However, in a multi-site scenario, the advantages of adaptive test in terms of test time compaction diminish since all sites must wait for the site that takes the longest test time. On the flip side, multi-site testing provides more information on each DUT since parameters of neighboring DUTs are spatially correlated. In this work, we take advantage of these neighborhood-based correlations to alleviate some of the challenges posed by multi-site restrictions on adaptive test. The proposed compound device approach enables us to incorporate device-to-device correlations of parallel neighbor devices. Moreover, we employ a neighbor statistics based defect screening mechanism to prune potential DPPM contributors. This neighbor-based defect screening helps us eliminate required tests that have been selected for defect screening. Using these two techniques, we achieve linear test time scaling with multi-site testing with respect to the number of sites in an adaptive test framework, while still providing the best test quality compared with prior work.

3. Adaptive Quality Binning for Analog Circuits

Per-device information based adaptive test selection enables us to greatly reduce test time while maintaining test quality level. Adaptive test selection mechanism described in previous sections target a fixed quality level and performs optimization according to that criteria. However, depending on the demand of customer, desired device quality level may vary.

Process variation creates a diversity in performance and quality of devices. The ones with higher quality are of higher value while the rest can be sold for a lower price. Separating manufactured devices according to their quality is defined as quality binning method and a very efficient way of lowering per-device cost. On one hand, devices of below average quality are not thrown away reducing per-device cost. On the other hand, above average devices are sold for higher prices.

Quality binned devices share the same design and typically go through same manufacturing and even the same test process. After the testing step, they are binned according to different sets of performance criteria. The bin a device falls depends on the manufacturing process and typically does not match the amount requested by the customers because of uncertainty of the process and arbitrary amounts of purchased devices.

3.1. *Methodology*

In our previous studies [Yilmaz and Ozev(2008), Yilmaz and Ozev(2009b), Yilmaz and Ozev(2009a), Yilmaz and Ozev(2010), Yilmaz *et al.*(2011)], we showed that there is a direct relation between DPPM and test time. Applying more tests reduces the expected number of defective escapes and therefore DPPM. The goal of this work is not only to minimize test time with a DPPM constraint as in the previous sections, but we speed-bin devices

such that the overall test time is minimized while the percentage of devices binned in each DPPM bin matches the demand. Problem formulation of this section can be summarized using the equations below:

$$objective: \quad min\, E[t] \tag{3.17}$$

$$constraints: \quad P(DPPM_i \leq DPPM < DPPM_{i+1}) = \quad P_i \tag{3.18}$$

$$\sum P_i \quad = \quad 1$$

where, E[.] is expected value operator, t and DPPM are random variables that represent test time and scaled defective escape probability, $DPPM_i$ are speed-bin limits and P_i is the proportion of the devices falling in the ith bin.

It is necessary to capture the relation between DPPM and test time in order to solve this problem. We use a conceptual figure to explain the main idea of the method. The relation between DPPM and time is illustrated in Figure (58) with a region instead of a curve to represent the statistical nature of the parameters. Note that uncertainty with respect to DPPM and time reduces as we apply more tests. This is intuitive because executing more tests enhance the confidence reducing the uncertainty to zero as the number of the executed tests approach to the maximum number of tests.

The mathematical tools presented in previous sections have been used to estimate the probability distribution of unmeasured specification parameters using measured specification parameters, $P(S_i|S)$. Where, S_i is the predicted specification parameter set and S is available specification parameter set. In this work, would like to incorporate DPPM and test time estimation into analysis. Therefore, we would like to estimate $P(S_i, D_i, time_i|S, D, time)$, where D is used as a shorthand notation of $DPPM$ and subscript "i" is the index of the ith DPPM bin specification. Once we establish the mechanism

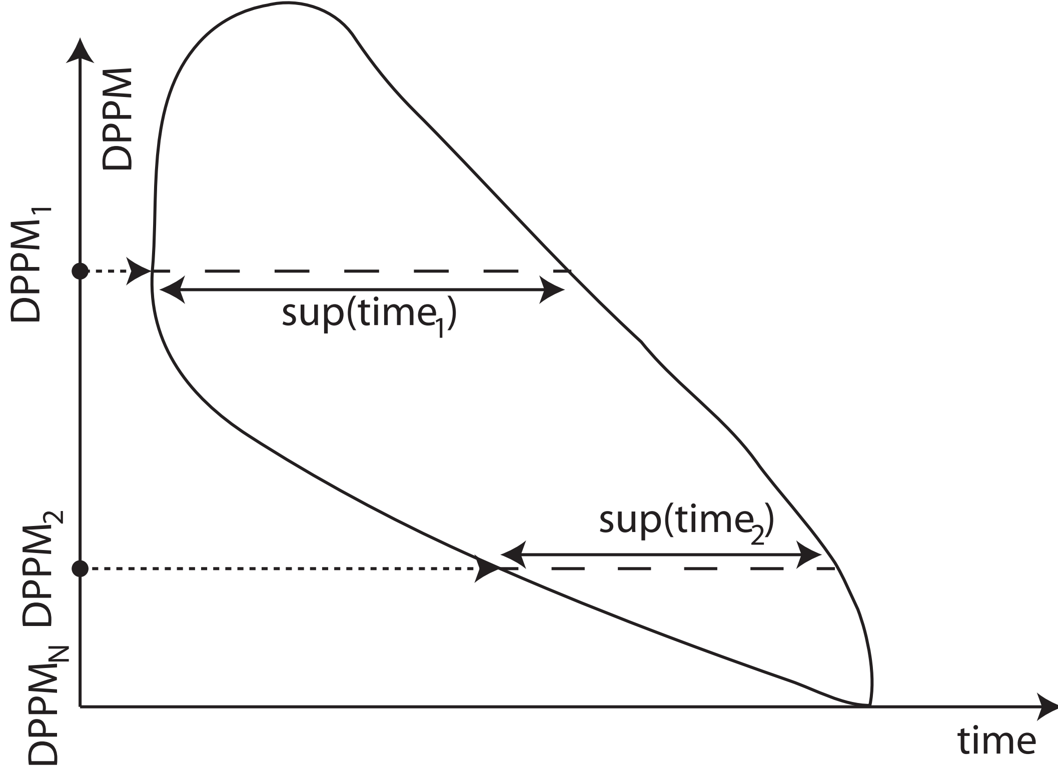


Fig. 58. Defective escapes per million (DPPM) and the number of executed tests are highly dependent. The dependency can be represented statistically.

to estimate this distribution the expected time for a particular DPPM level can be estimated and a decision rule can be obtained for binning to yield minimum test time. This process is illustrated in Figure (58). Joint distribution is sliced at the desired DPPM levels to estimate test time distributions. The main idea in this work is to estimate these distributions and put devices in bin#i if a device fall in the lowest P_i of its $P(time_i)$ distribution.

We use equation (3.19) to estimate the expected test time.

$$P(t_i|S_i, D_i, S, D, t) = \frac{P(S_i, D_i, t_i|S, D, t)P(S, D, t)}{P(S_i, D_i, S, D, t)} \tag{3.19}$$

Integrating the equation with respect to S_i and D_i yields the distribution of test time:

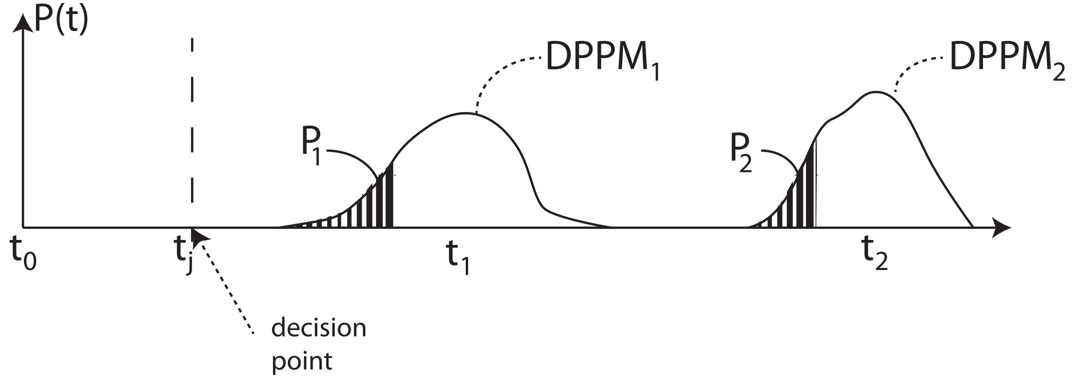


Fig. 59. The number of test required to achieve a DPPM level can be represented using a statistical distribution. Devices falling in the lower tail of the statistical distribution of test time parameter is used to minimize test time.

$$P(t_i) \quad = \int \quad P(S_i, D_i, t_i | S, D, t) dP(S_i, D_i)$$

Statistical binning process is illustrated in Figure 59 in more detail. Once a measurement set becomes available, it can be used to estimate test time distribution of a device for bin#i ($DPPM_i$). However, we execute the decision only at critical points where a decision is necessary, where the decision points are DPPM bin boundaries.

During the initial phase of testing a device, uncertainty is large and a number of tests are required to reach the first quality bin boundary $DPPM_B$, where B is the total number of bins. At this point we need to decide whether we should put the devices in the lowest quality bin and stop testing or if we should put the device in another bin ($DPPM_i$) and continue testing until $DPPM_i$ us maintained. The algorithm of the process is listed in Figure (60). Decision mechanism is executed when DUT reaches a DPPM boundary. At $DPPM_j$, if the device under test falls in P_i of the lower tail of the estimated $P(t_i)$, where

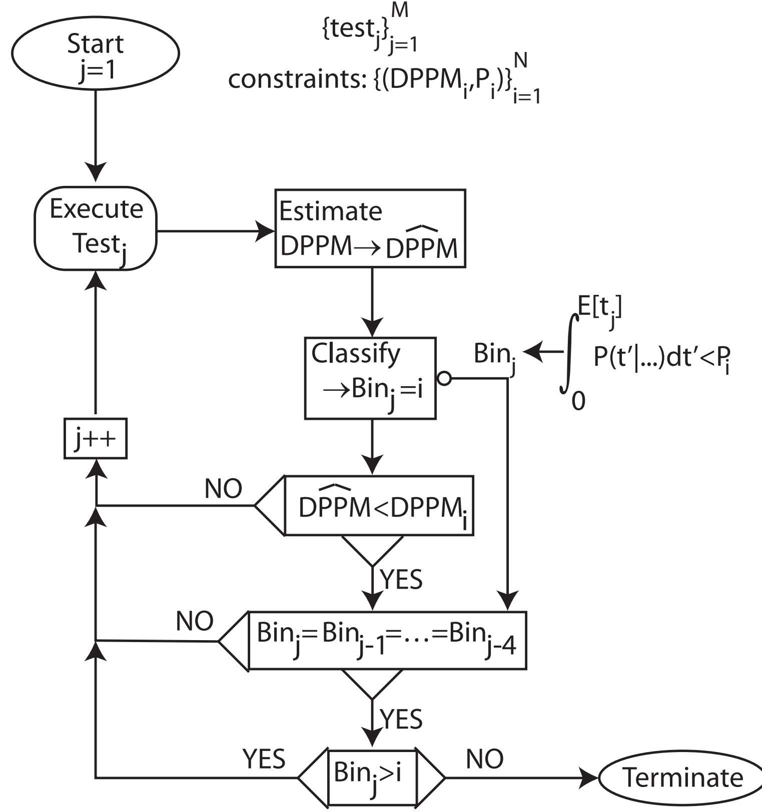


Fig. 60. Adaptive qualitybinning test flow

$i > j$, the DUT is put in bin#i and continued to be tested. Otherwise, the device remains in bin#j and testing is stopped.

3.2. *Training Det Dize Related Stability*

Although the method discussed above is mathematically sound, it is based on the asymptotic properties of estimation and may not be stable for small training sets. The equations used to estimate $P(t_i)$ assume an infinite learning set and therefore may not yield the best results for limited training sets. The importance of decision stability can be best explained through an intuitive example. Suppose that the binning algorithms has reached a decision point j at $time_j$ and decides to put the DUT to bin#i ($DPPM(DUT) < DPPM_i$). We

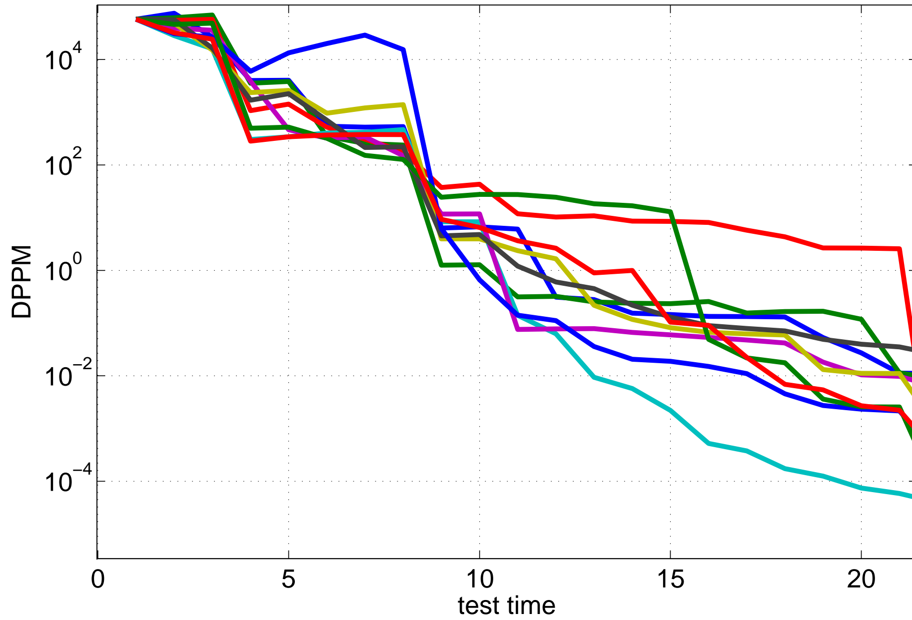


Fig. 61. Defective escape probability reduces with the number of applied tests. However the estimated parameter cannot be guaranteed to be monotonically non-increasing due to the limited size of learning data set.

not only want the decision to be estimated but also to be a stable decision within the neighborhood of decision point j. That is, if the decision at point j-1 or at j+1 is different, or alternates frequently, it is not stable. We define stable decision as decision remains at a specific bin and does not change very frequently.

We demonstrate the stability issue using Figure 61. Estimated DPPM parameters is shown with respect to time. The figure shows that estimated DPPM value of some devices change more rapidly than the others. This can be attributed to poor characterization due to limited training set size and excursions in some measurement parameters that de-stabilize the estimation mechanism.

To improve the stability of the algorithm, we add an additional constraint. Although decision is performed at bin boundaries, we generate a bin estimation B_k at every test, where k is test index. In order to finalize a decision we require 5 consequent classifications of the same kind.

3.3. *Results*

We evaluate the performance of the proposed method and compare it with other work using two sets of production data. The data is collected from production line through exhaustively applying the whole test suite. The first data set has 42 specification parameters of a large scale mixed-signal circuit of 89k devices. The second data set has 264 specification parameters and 900k devices and large scale analog circuit.

Quality binning simulations are performed using training set size of 2000 devices and repeated 10 times to obtain a stable estimate of the performance. The methods that are used for comparison are subjected to the same averaging process. Test time of the individual tests are assumed to be unity. Hence, test times reported below correspond to the number of applied tests.

We demonstrate the performance of the proposed method using two sets of constraints listed in Table 13. The purpose of using two sets is to show that device binning can be controlled using our method.

3.3.1. *Data Set#1.* Figure 62 splitting performance for the first data set using 5 runs with disjoint training sets of devices. Figure 62(a) and (b) show results for the first and second constraint respectively. Blue dashed lines show the ideal splitting level, while red lines represent simulation results. Split percentages almost overlap with the ideal splitting conditions. Therefore conditions are satisfied.

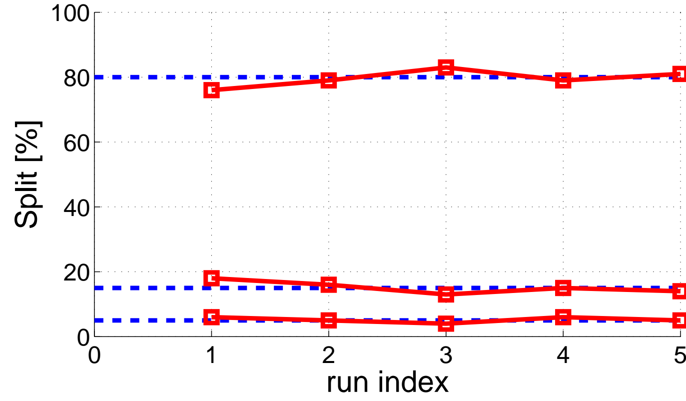
DPPM Constraint	Percentage Constraint Set#1	Percentage Constraint Set#2
500<DPPM<1000	85%	70%
0<DPPM<500	15%	20%
DPPM=0	5%	10%

Table 13. Binning constraints

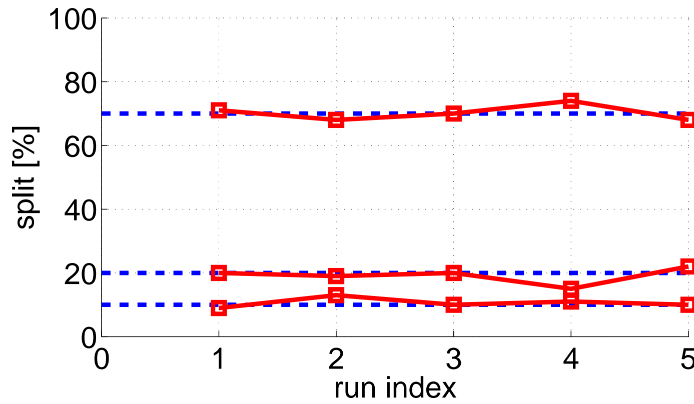
Fluctuation of the red curve is due to the limited size of the training set size and potential shifts in the process. Under represented or invalidated statistical characterization data may result in inaccurate splitting ratios.

In the figures above, we showed that the constraints are met with acceptable performance variations. Now, we show test time reduction performance of the quality binning method. Figure 63 shows the results and performance of the state of the art methods on the same plot. Isolated marked points show performance of the other methods. Since most of these methods do not have a DPPM control mechanism, they occupy a single point on the plot. We used a test covering based method [Yilmaz *et al.*(2011)] to show the relation of DPPM-test time, illustrated using the solid blue curve. Data points of this curve are obtained by changing the size of a test list generated using incremental coverage criteria. This helps to understand the dependency between DPPM and test time.

Performance for the first constraint set is shown with the vertical dashed red line that has three markers. Vertical location of the markers correspond to the binning criteria. Split rate of each condition is marked next to the markers. Horizontal location of the markers indicate test time performance. For comparison. traditional cover based method is



(a)

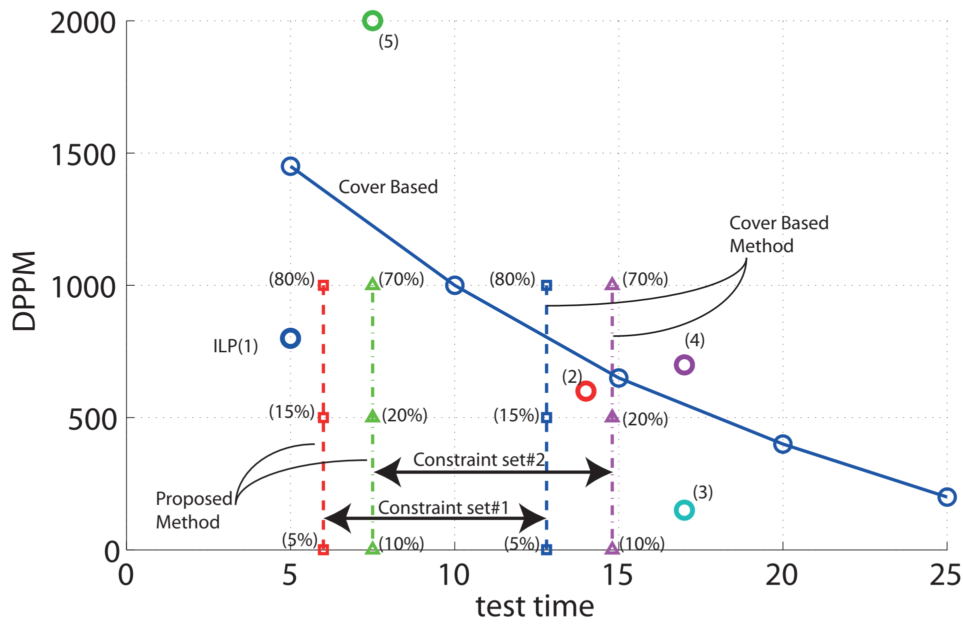


(b)

Fig. 62. Binning percentages for data set-1. Splitting ratios almost overlap with the target levels

evaluated with the same condition set (#1) and results are plotted using a blue dashed curve on the same figure. The figure shows that the proposed method reduces quality-binning time to 50% compared to the traditional method.

3.3.2. *Data Set#2.* Splitting performance is evaluated and illustrated in Figure 64 for the second data set. The results show that splitting performance is satisfactory despite of the fluctuations in performance.



(1) [Stratigopoulos *et al.*(2007)]

(2) [Milor(1998)]

(3) [Chen and Orailoglu(2008)]

(4) [Biswas *et al.*(2005)]

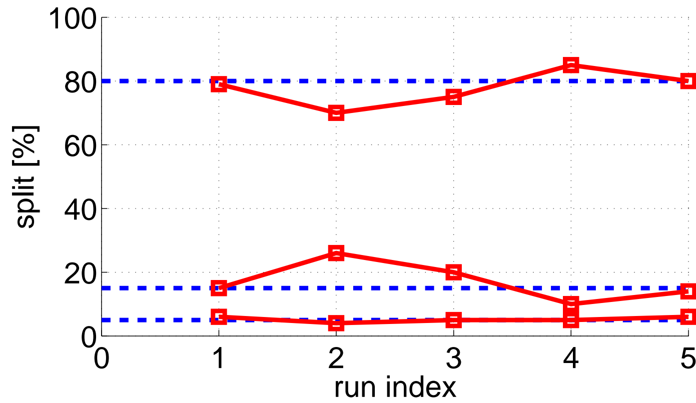
(5) [Benner and Boroffice(2001)]

Fig. 63. Vertical dashed lines show the results for binning simulation. Each marker on the vertical lines correspond to a binning criterion. Split ratio of the bins are shown next to the marked point.

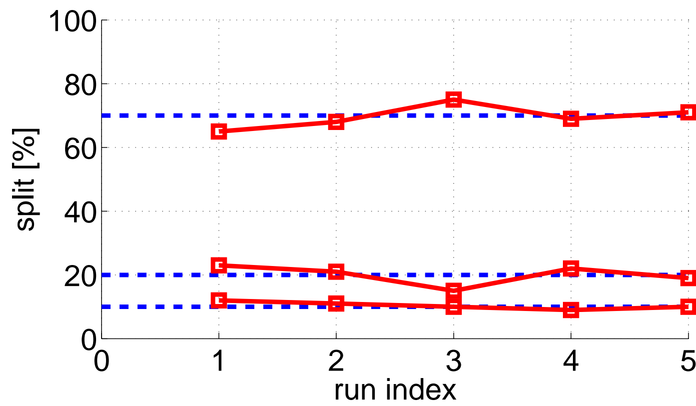
Test time performance of the second data set is illustrated in Figure 65. Our proposed method achieves more than 60% test reduction for both of the constraint sets.

4. Multidimensional Outlier Detection

Outlier devices behave differently from the majority of the devices and are considered to be potentially defective. Identifying outliers has many applications in test, including defect



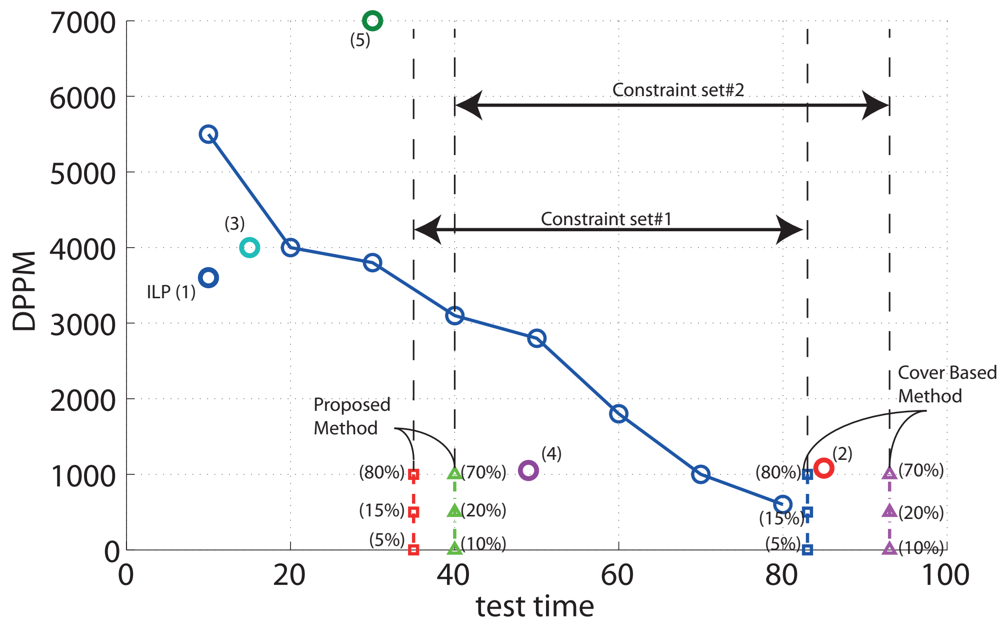
(a)



(b)

Fig. 64. Binning percentages for data set-1. Splitting ratios almost overlap with the target levels

filters for alternate test, and setting pass/fail limits for automotive domain. In previous work, outliers have been identified using single dimensional and/or static methods which does not exploit information efficiently. In this work, we propose an adaptive multidimensional outlier analysis method that combines the information of multiple measurement parameters and judiciously selects only information rich parameters to maximize detection



(1) [Stratigopoulos *et al.*(2007)]

(2) [Milor(1998)]

(3) [Chen and Orailoglu(2008)]

(4) [Biswas *et al.*(2005)]

(5) [Benner and Boroffice(2001)]

Fig. 65. More than 60% test compaction is achieved for the given constraint sets.

probability. Furthermore, the proposed method continuously updates to track process shift to enable adaptation to the evolving processes.

The proposed method can be integrated within an existing test framework to improve test quality with little or no additional test time cost. In this context, we integrate our technique with an adaptive test framework and show that the method improves test quality.

4.1. *Background*

Outlier is typically defined as an observable deviation from the typical behavior. This observable change can be attributed to physical defects or an unpredictable deviation of the process altering the behavior of the affected device. Outlier concept can be best explained through an illustration. Figure 66(a) shows a scatter plot of two parameters for several identical devices. Circles represent defect free devices, cross signs represent defective devices, and the dashed square shows specification boundaries. Note that if only specification boundaries are used in the analysis, only the leftmost defective circuit is detected. However, there are several more devices that behave significantly different from the typical circuits (shown with cross signs). All of these suspicious devices needs to be identified.

However, identification of outliers is not trivial. Process variations introduce uncertainty into the identification process, masking the deviation of some defects. For example, circled defective devices can be relatively easily detected in Figure 66(a), however, uncircled ones are very close to typically behaving devices, therefore, are hard to detect. Several methods have been proposed to achieve a good outlier identification rate.

4.1.1. *Outlier Analysis.* The most easily implemented and the least effective outlier analysis method is one dimensional approach, where device parameters are analyzed separately. Figure 66(b) shows a typical 1-D outlier analysis method for parameter Y in Figure 66(a). First, the defect free distribution of the devices is generated, which is represented with a Gaussian curve in the figure. Then, outlier boundaries are generated to separate

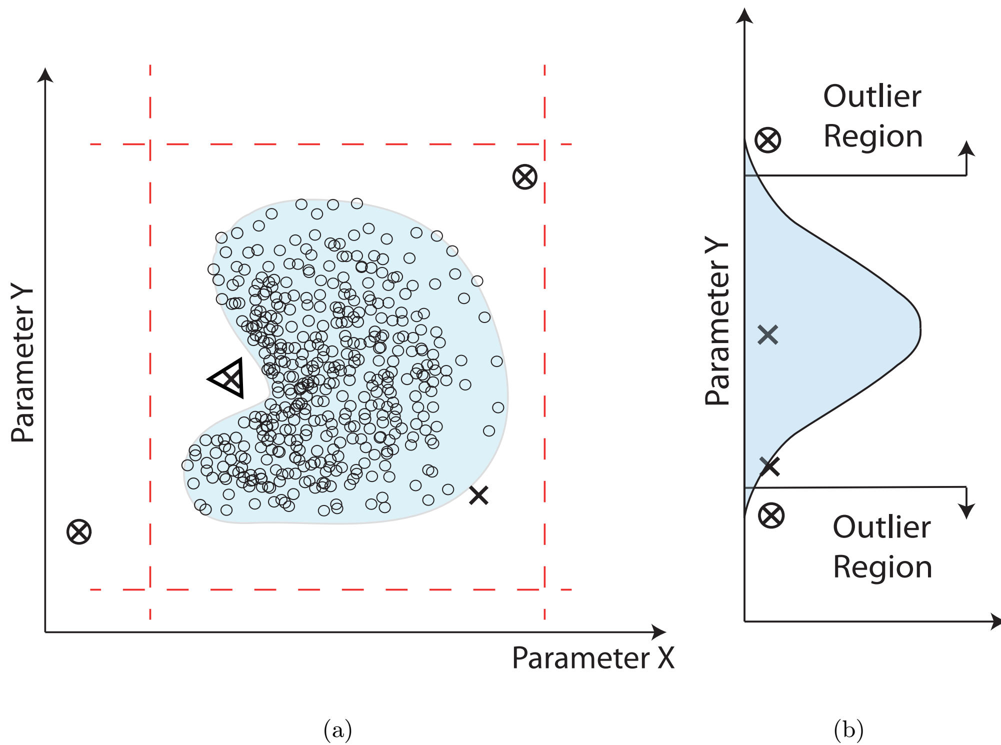


Fig. 66. Outliers can be detected using multiple dimensional analysis. Outliers can be detected easier in (a) 2D than in (b) 1D

the region of defect free devices from the region of potentially defective devices. Direct and indirect measurement parameters can be used in this analysis.

However, 1-D approach is limiting and inefficient in outlier identification. For example, consider the potential defect encircled with a triangle in Figure 66(a). It cannot be detected neither in X dimension nor in Y dimension with 1-D approach. But, it can be much easily detected using a 2-D approach as shown in the 2-D scatter plot. Therefore, performing outlier analysis in higher dimensions would conceivably improve the identification success.

In [Yilmaz and Ozev(2009b)] we presented a low cost 2-D approach to improve outlier identification rate. However, this approach cannot be generalized to N-dimensions due to

the difficulty in representing outlier decision boundaries in multiple dimensions. A nonlinear boundary generation method is proposed in [Stratigopoulos and Makris(2005)] using neural networks, but its is not suitable to be used in high dimensional spaces.

4.1.2. *Multi-D Outlier Analysis.* Multi dimensional outlier analysis methods are proposed in [Cerioli(2009), Filzmoser *et al.*(2005), Papadimitriou *et al.*(2003), Pena and Prieto(2001)]. But, they either use parametric models to model parameter distributions or employ cluster-based [Jiang *et al.*(2001)] representation of instances. These approaches cannot be used to represent complex distributions of analog/mixed-signal circuit responses or computationally costly to use with a large number of dimensions.

A more sophisticated nonparametric approach is proposed in [Stratigopoulos *et al.*(2009a)] to pre-filter outliers for accurate model generation. However, updating statistics adaptively and uncertainty reduction are not considered in this outlier analysis methods.

As we will show in the results section, these two issues are very important in outlier detection.

4.2. *Proposed Approach To Analyze Outliers*

We use a similar concept used in multi-D outlier methods [Cerioli(2009)] to shrink the number of dimensions. We first explain a simpler approach where device parameters are assumed to have Gaussian distribution and then explain how we extend the method with nonparametric models.

Suppose that device parameters are Gaussian with mean vector μ, and covariance matrix C. Then, Mahalanobis distance is defined as;

$$MD^2(s_i) = (s_i - \mu_i)^T C^{-1} (s_i - \mu_i) \quad (3.20)$$

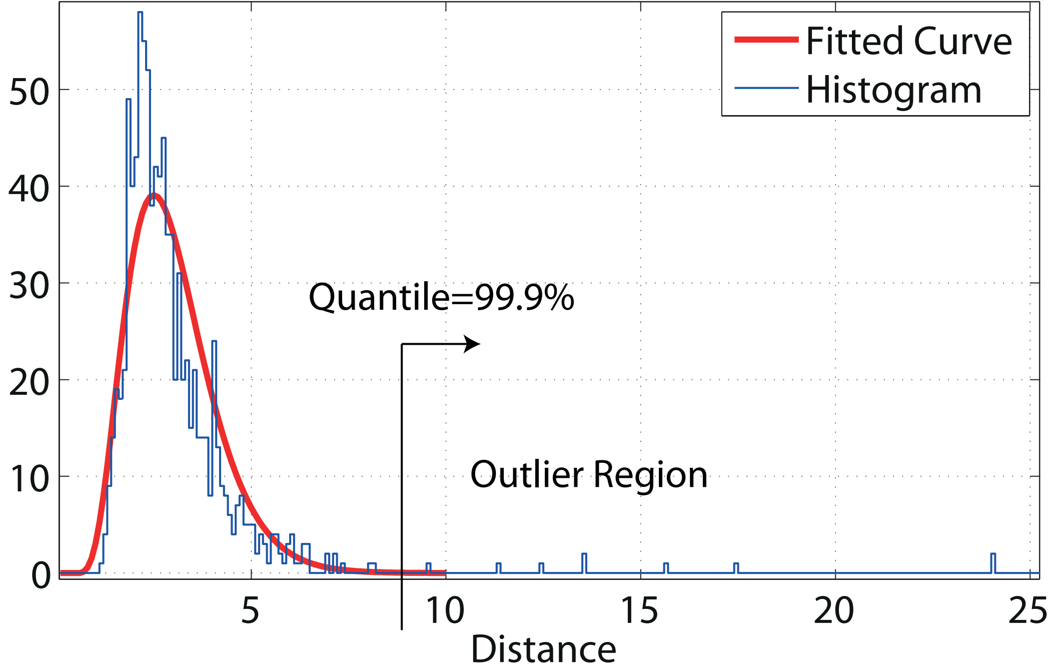


Fig. 67. Profile of the distribution of $D_{\{j\}}(s_i)$ can be modeled parametrically

where s_i is measurement parameter vector of ith device, and $(.)^T$ is transpose function. This transformation effectively enables to transform multidimensional parameters into a one dimensional distance parameter greatly simplifying analysis.

Unfortunately device response distributions are much more complex due to the non-linear nature of circuits with respect to their inputs. Hence, Gaussian assumptions are not typically satisfied. Instead, we employ a nonparametric modeling method and define distance measure as a natural extension of this model.

We employ a kernel based non-parametric method to represent joint probability distribution function of device parameters. A more detailed explanation of the method can be found in [Yilmaz and Ozev(2010), Scott(2008)]. Distribution of the model is generated by placing a kernel function in the high dimensional space for a sample characterization set of defect-free devices. Distribution is constructed using equation (3.21).

$$pdf(s_i) = \frac{1}{N}\sum_{k}^{N}\prod_{j}^{M} K_j(\frac{s_{i,j}-\mu_{k,j}}{h_j}) \quad (3.21)$$

where N is the size of the characterization set, M is the total number of dimensions, $s_{k,j}$ is the measurement of jth parameter of ith device, K_j is the smoothing kernel for j^{th} parameter, while $\mu_{k,j}$ and h_j are mean and kernel width parameters for j^{th} parameter respectively. This equation enables to fit a smooth multidimensional function to represent the parameter distribution of even nonlinear analog devices. Now we define a distance measure that effectively shrinks the number of dimensions to one. The order of summation and product in (3.21) can be interchanged if the multidimensional kernels instances are sufficiently localized, or uncorrelated. This leads us to the following distance measures obtained through logarithmic transformation.

$$D_j(s_i) = -log\left\{\frac{1}{N}\sum_{k}^{N} K_j(\frac{s_{i,j}-\mu_{k,j}}{h_j})\right\} \quad (3.22)$$

$$D_{\{j\}}(s_i) = \sum_{\{j\}} D_j(s_i) \quad (3.23)$$

D_j is the distance of device parameter vector s_i in j^{th} dimension. $D_{\{j\}}$combines the distances defined by D_j in parameter set $\{j\}$ and yields a scalar number that is the distance of s_i from the nominal. This scalar number summarizes the deviation of a device from the nominal and enables to easily incorporate information of multiple dimensions into outlier analysis. Note that, $D_{\{j\}}(s_i)$ reduces to Mahalanobis distance (with identity covariance matrix) when N=1, the distribution of device parameters is Gaussian.

Outliers can be detected by analyzing the location of devices in one dimensional distance ($D_{\{j\}}$) space. For the simpler case, where device parameter distribution can be assumed to be Gaussian, $D_{\{j\}}$ has χ^2 (Chi-Square) distribution. As a result, $D_{\{j\}}$ is the overall

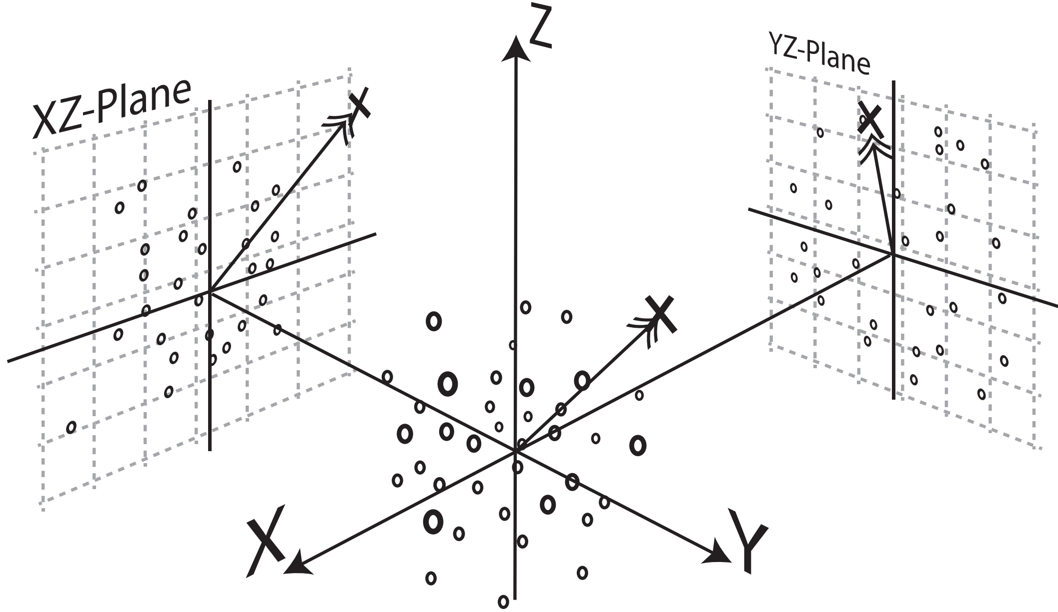


Fig. 68. Not all parameters bear information. Using only information containing parameters enables to achieve a better detection rate

normalized distance of devices from the nominal. Outliers can be identified by assigning a boundary at a high quantile and identifying devices as outliers if they fall beyond that level.

For the nonparametric model employed in this work, it is hard to derive the statistical distribution $D_{\{j\}}$. Despite the complexity of the model, $D_{\{j\}}$ closely resembles a χ^2 distribution. Hence χ^2 distribution can still be used in our model. Illustration in Figure 67 shows how well the fitted distribution matches the experimental data. In the figure, blue bars represent the histogram of distance parameter while the red curve is the fitted a parametric model. Note that we are especially interested in fitting the right tail of the distribution and a good fit is achieved using the parametric model.

Then we select a point using the fitted model using a high quantile (inverse of cumulative distribution function), such as 99.9%, on the distance space to separate outlier region from

the outlier-free region. The separation of regions is shown in Figure 67 using a broken arrow. Quantile level can be used to set the trade-off between yield loss versus detection rate, where yield loss is approximately $(100 - quantile)\%$.

4.3. *Uncertainty Reduction*

Using combined information of multi dimensional data yields more information and enables us to make a better informed decision. However, this does not necessarily mean that all measurement results have to be used to maximize information. On the contrary, the measured parameter set needs to be carefully selected. For example, a defect may affect only a subset of specification parameters while leaving the others unchanged. Utilizing the information of only altered parameters would be sufficient in the analysis. If all parameters are used, uncertainty introduced by unaltered, information-poor parameters may result in an incorrect decision. This can be best explained with an example. Figure 68 shows scatter plot of a conceptual example in three dimensions for specification parameters X, Y, and Z, where hollow circles represent defect free devices, and cross sign represents a defect. Projection of the data on XZ plane and YZ plane are generated to show individual contribution of parameters. In this example, projection on XZ plane shows that X and Z parameters bear enough information to deem the defect as an outlier. Now suppose that we would like to incorporate Y parameter into the analysis. The projection of data on YZ-plane shows that the defective device is located very close to the mean in Y dimension and is almost indistinguishable from defect free devices. Therefore if Y parameter is used, no new information is obtained, while additional uncertainty is introduced.

To eliminate uncertainty induced by such parameters, we need to decide which subset of parameters to use in outlier analysis. We derive equations of a simplified model to

gain a better understanding of the problem. Suppose that device parameters, s_i, can be represented with the model given in Equation (3.24).

$$s_{i,j} - \mu_j = x_{i,j} + w \tag{3.24}$$

where $s_{i,j}$ is the ith specification parameter of the jth device, w is uncertainty introduced by process variation and measurement system, μ_j is the nominal value, while $x_{i,j}$ is the deviation imposed on i^{th} device by the defect in the j^{th}device parameter. Uncertainty term, w, is assumed to be independent and identically distributed for all device parameters ($w = w_{i,j}$). In order a defect to be observable, it should manifest itself as a significant change to be distinguishable in the presence of uncertainty. We define a measure to describe the concept more clearly.

$$DUM_{\{j\}}(s_i) = \frac{\sum_{\{j\}} E[x_j^2]}{L \cdot E[w^2]} \tag{3.25}$$

where DUM is detection utility metric of the distance measure, L is the size of test list {j}, and E[.] is the expected value operator. In this equation, DUM represents the information content and is desired to be large. In order to obtain a large DUM, the subset of test set, {j}, to be used in outlier analysis should be carefully selected to maximize DUM.

We use a heuristic approach, shown in Algorithm 3.1, to select the information containing measurement set $j_{opt(s_i)}$ from all measurements $\{j'\}_1^{M'}$, where M' is the total number of available measurement parameters. We first select a test j such that it yields the maximum distance parameter ($D_j(s_i)$) according to the equation (3.22) and append it to test set $\{j\}$. Then, we calculate the DUM of test set $\{j\}$ using equation (3.25), where $E[w^2]$ is calculated using cumulative statistics and $E[x_j^2]$ is simply D_j. We keep adding tests to $\{j\}$

Algorithm 3.1 Uncertainty Optimized Test List

$\{j\}_{opt} \leftarrow \{\}$; $maxDUM = 0$

while 1 **do**

$m \leftarrow argmax_n(DUM_{n:n\in\{j'\}_1^{M'}-\{j\}_{opt}}(s_i))$

if $DUM_{\{j\}_{opt}+\{m\}}(s_i) < maxDUM$ **then**

return $\{j\}_{opt}$

else

$\{j\}_{opt} \leftarrow \{j\}_{opt} + \{m\}$

end if

$maxDUM \leftarrow DUM_{\{j\}_{opt}}(s_i)$

end while

using until the computed DUM stops to increasing. The generated test set is used as the most informative test set for the device under test.

4.4. *Integrating Outlier Analysis Into a Test Flow*

Figure 69 shows the flow of a test approach that uses the proposed adaptive outlier analysis technique. A typical test flow includes an initial test list generation step, and a Test Engine (testing/evaluation) step. These steps are illustrated in the gray region in the Figure 69. The proposed method, shown with the yellow region in the figure requires two modifications in the test system: placing an outlier analysis step and an statistics-update step after the test engine (pass/fail evaluation step), and appending a set of tests into the initial test list. Note that the test engine can be any testing method, such as set cover, alternate test or even IDDQ testing. In this work, we use an adaptive test method [Yilmaz and Ozev(2010)] as the test engine and embed the proposed method in the flow as suggested.

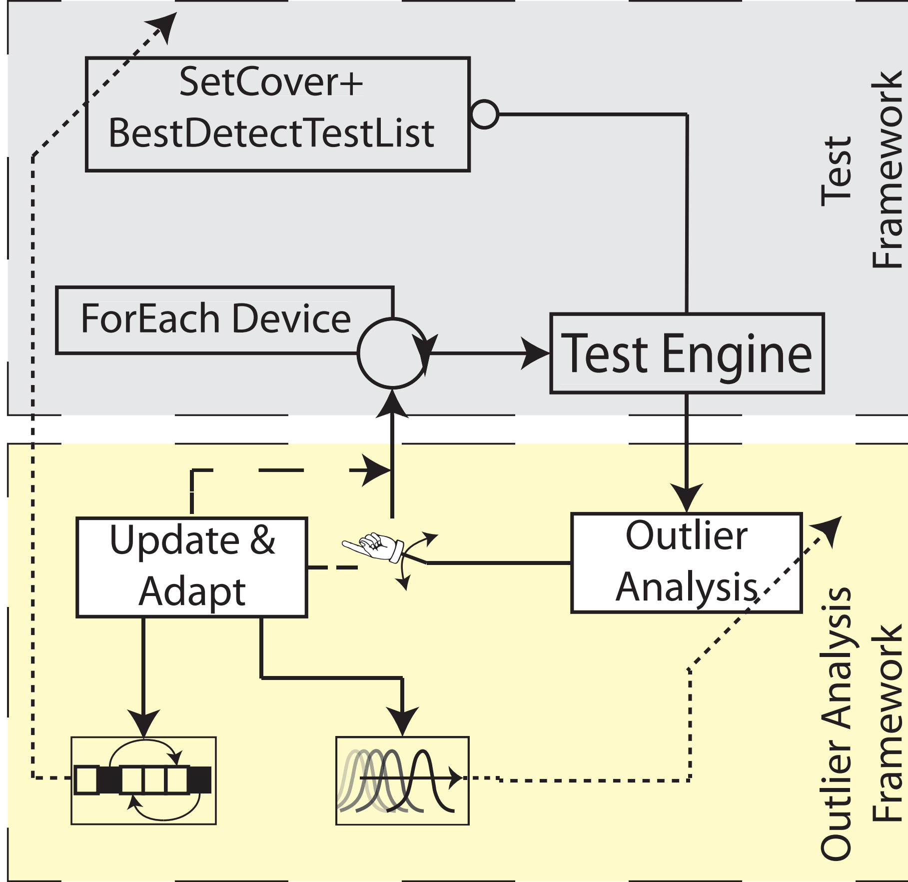


Fig. 69. Adaptive outlier analysis flow

4.5. *Determination of Initial Test List*

Achieving the most information and therefore maximum DUM depends on the available measurement parameter set, which strongly depends on the initial test list. We propose a method to generate a subset of most informative tests in order to increase the chances of detecting outliers. The generated test list can be simply appended to the initial test list that is generated by the host test system. In our work, we simply replace heuristically generated parts of the initial test list with the proposed method.

First we analyze a sample set of devices, $\{t_i\}$, to learn the statistics of the system. We guarantee to use defect free devices in the analysis by filtering out potentially defective

devices very aggressively. We use robust estimators median and MAD (maximum absolute deviation) to accept only 4σ neighborhood of the parameters, yielding $\{t'_i\}$. Once the potential defects are aggressively filtered from the training set, we use it to compute statistics such as $E[w^2]$ and parametric distribution of distance parameter $(D_j(t_i))$, which are sensitive to defects. Then we compute the most informative test list for each device in $\{t_i\}$ and compute their quantile according to the parametric distribution generated using $\{t'_i\}$. The devices that exceed a certain level of threshold distance are identified as outliers and put into a set $\{t_i\}_{out}$. Now, we determine a common test set to each device in $\{t_i\}_{out}$ that can successfully identify all of them.

We use a heuristic approach and incrementally add a test that achieves the most outlier coverage. The procedure is listed in Algorithm 3.2. The algorithm stops when all devices in $\{t_i\}_{out}$ are identified and yield the best common detection test list, $\{j\}_{BD}$.

Adapting capability of the outlier analysis requires minute changes in the test engine, such as appending a few more tests to increase detection probability. But, it is important to keep these changes at an acceptable level especially if the number of applied tests is an important concern for the test engine. In this work, we replaced intuitively generated test list in [Yilmaz and Ozev(2010)] with our initial test list generation method and the size of initial test list remained more or less the same. Therefore, test time overhead is almost zero.

4.6. ***Adaptive Outlier Detection***

Outlier analysis step is the successor of the Test Engine step, shown in Figure 69. In Figure 70, we show a more detailed diagram of outlier detection process. Test engine conducts some tests and decides whether the DUT (device under test) passes or fails.

Algorithm 3.2 Generate best detecting test list

$\{t'_i\} \leftarrow Filter\{t_i\}$

$E[w^2]$,$D_j(\{t'_i\})$),$PDF(\chi^2) \leftarrow$ Estimate Using $\{t'_i\}$

$Th = Quantile(\chi^2, 99.9\%)$

$\{t_i\}_{out} \leftarrow \{\}$

for all $t_i \in \{t_i\}$ **do**

  if $\chi^2(t_i) > Th$ **then**

    $\{t_i\}_{out} \leftarrow \{t_i\}_{out} + t_i$

  end if

end for

$\{j\}_{BD} \leftarrow \{\}$

for all $i \in \{i\}_1^M$ **do**

  $detectRate_{best} \leftarrow 0$

  for all $j \in \{j\}_1^M - \{j\}_{BD}$ **do**

    if DetectRate($\{t_i\}_{out}$,$\{j\}_{BD}$)$>$$detectRate_{best}$ **then**

      $j_{best} \leftarrow j$

      $detectRate_{best} \leftarrow$ DetectRate($\{t_i\}_{out}$,$\{j\}_{BD}$)

    end if

  end for

  $\{j\}_{BD} \leftarrow \{j\}_{BD} + j_{best}$

  if DetectRate($\{t_i\}_{out}$,$\{j\}_{BD}$) $=$ 100% **then**

    return $\{j\}_{BD}$

  end if

end for

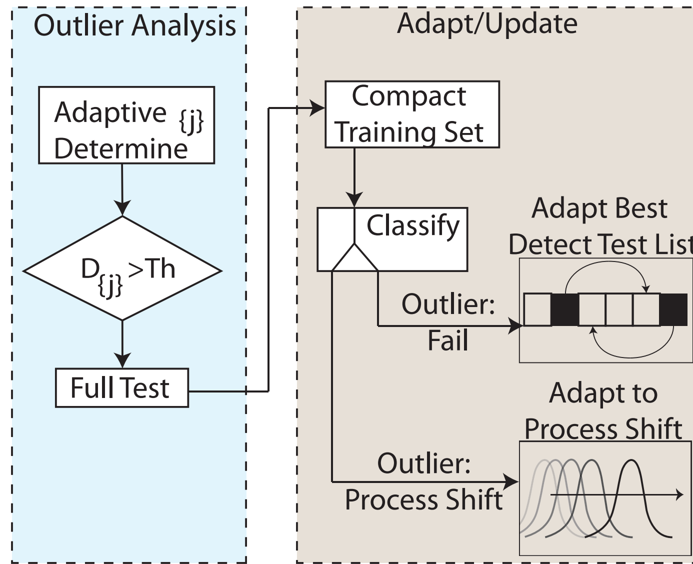


Fig. 70. Outlier analysis flow

Outlier analysis step, uses the measurement results obtained in the previous step $\{j'\}$ to identify whether the DUT is an outlier or not.

First the most informative test list $\{j\}_{opt(s_i)}$ is determined using the algorithm provided in section 3.1. Then the distance of the device $(D_{\{j\}}(s_i))$ is computed using equation (3.23) and compared to a threshold level that corresponds to a high quantile level, such as 99.9%. If the distance of the DUT exceeds the threshold value, it is subjected to exhaustive testing. Otherwise, outlier analysis stops and the control is passed to the test engine to continue with the next device. Exhaustively tested potential outlier devices also enables us to update process statistics on-the-fly and therefore adapt to the process shift.

4.7. *Updating/Adapting to Process Shift*

Adaptive update of statistics are performed periodically every several thousand devices due to the computational overhead of the update process. We chose this period to be 10k in our analysis. Potential outliers that have been determined during the test flow are analyzed

to update the process statistics. Instead of re-characterizing the process by exhaustively testing a large number of devices, we utilize already measured device information. Process shift results in a change in process statistics. This shift manifests itself as identification of more potential outlier devices. Therefore we use some of the potentially defective devices obtained in the outlier analysis step to update representative set of kernels and the rest to update the best detection test list. Some of the devices may be wrongly classified as outlier devices due to outdated process statistics information.

We employ a compaction mechanism to classify these devices into two groups. The compaction algorithm simply combines aggregated device instances in the multidimensional parameter space. If a group of devices are in proximity of a certain radius they are joined into a device group. Since we would like to determine which potentially defective devices are actually defective and which are a result of a process shift, we apply the compaction algorithm on the outlier devices and determine if they combine or remain individual. Combination of the devices implies that some outliers are in close proximity. According to the definition of outliers, they should behave differently from the majority of the devices and we expect them to be rather scattered instead of being concentrated. Hence, devices that combine indicate a possible process shift. We select devices that combine in groups of 10 or larger to update the training set list. This enables to adapt the model of the outlier detection method to the changing process conditions. The devices that do not combine, however, are used to update the best detection test list.

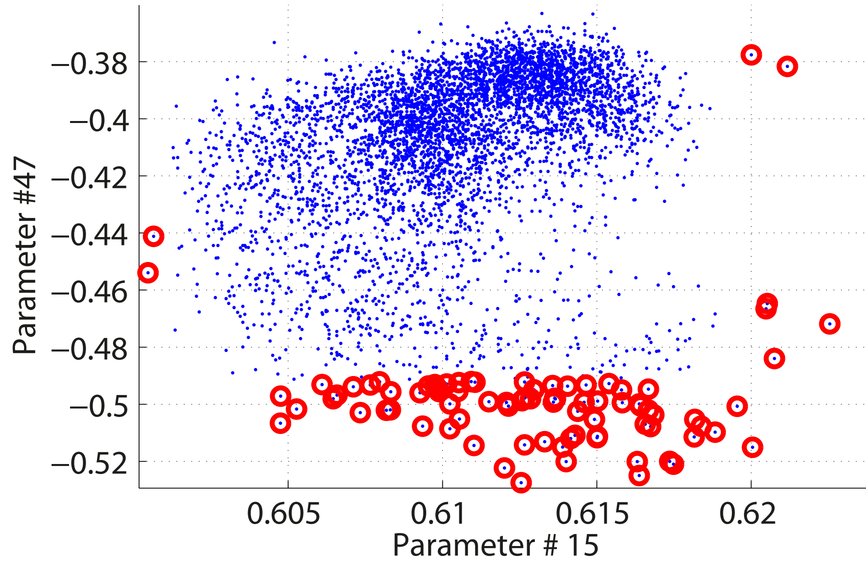
4.8. *Results*

We apply the proposed method to three different sets of industrial data of large scale mixed-signal circuits. The first data set has 42 specification parameters for a large scale

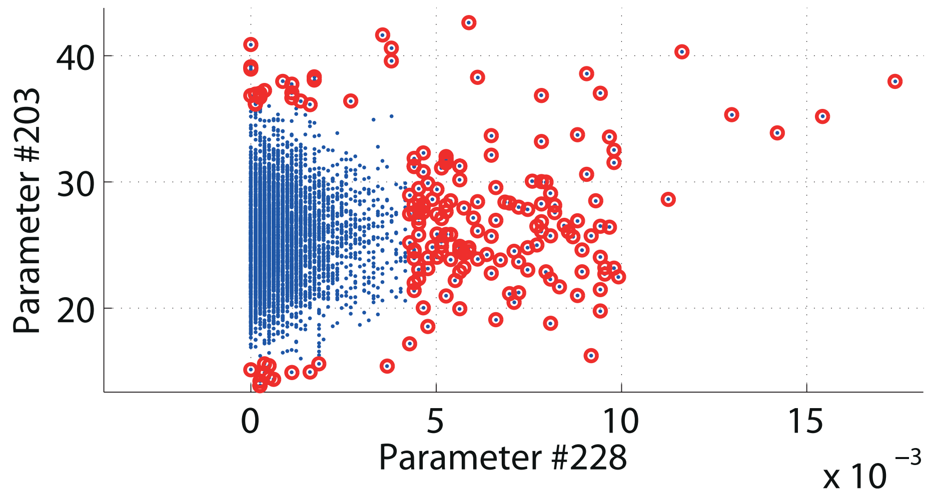
mixed-signal circuit of ~89k devices. The second data set has 21 specifications of a small scale analog circuit of ~21k devices. The third data set is of a large scale circuit with 264 specifications and ~900k devices. Throughout the analysis we use disjoint training sets of 2000 devices for characterization.

4.8.1. *Outlier Analysis Efficiency.* First we illustrate the performance of the method in two dimensions to see how well outliers are identified visually. In Figure 71(a) a 2D scatter plot of parameter #15 and parameter #47 are shown for the third data set. Red circles represent the devices identified as outliers. The figure shows that the proposed method is capable of identifying devices behaving even slightly different from the majority. Figure 71(b) is another 2D illustration for randomly picked device parameters for the same data set. Identification capability is visually verified in this figure.

We showed that information in multiple dimensions can be extracted through an appropriate transformation of multiple dimensioned data to a single dimensional distance metric. However, in section 3.1 we argued that only information rich parameters should be used to maximize DUM and therefore detection probability. Furthermore, shifting process statistics needs to be tracked. Now we show how these two can substantially improve outlier identification capability. We compare the results of our method with a multi dimensional method [Stratigopoulos *et al.*(2009a)]. Algorithm based on [Stratigopoulos *et al.*(2009a)] is integrated into a cover based test engine and the distance measure defined in equation (3.23) is used for comparison. First we show the effect of uncertainty reduction in outlier detection. In Figure 72(a), we compare the distance metric for the case where all available information is used (method based on [Stratigopoulos *et al.*(2009a)]) and where only information containing parameters are used in outlier analysis. A section of distance metric for device instances with index of 720 to 820 are shown to prove our point. The red curve



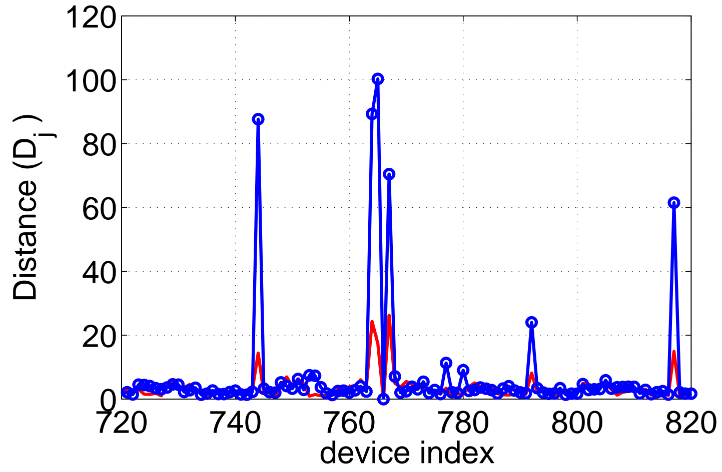
(a)



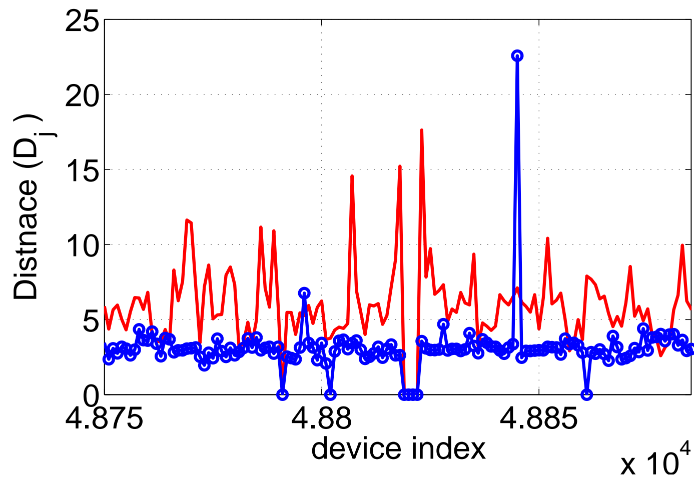
(b)

Fig. 71. 2D visualization of parameters illustrate the operation of the method

shows the distance metric($D_{\{j\}}(s_k)$) for device k with no uncertainty aware optimization, and blue curve shows distance metric of the same device set with uncertainty aware test set selection. The spikes in the figure indicate outliers and therefore potential defective circuit.



(a)



(b)

Fig. 72. (a) Device distance, is more significant when uncertainty aware test selection is performed. (b) Adapting to the process is necessary to guarantee consistent and reliable outlier identification.

Note that spikes of the red curve are very close to the minimum detection level and cannot be distinguished clearly. However, blue spikes are clearly observable. This discussion

suggests that using information of multiple dimensioned data may not provide the most information if uncertainty aware analysis is not performed.

Now we turn our attention to the effect of tracking the process shift. The effect of process shift in distance parameter is illustrated in Figure (b). The plot is generated after simulating ~48k devices. The blue curve is the results of our method and the red curve is the results of the method based on [Stratigopoulos *et al.*(2009a)], where process shift is not tracked. The figure shows that statistics of the static method is shifted upward and standard deviation is increased. Therefore, sensitivity of the method is diminished. Although the method works well initially (shown in Figure 72(a)), process shift invalidates the process information (show in Figure 72(b)). Furthermore, results are not reliable anymore since decisions are based on outdated information. Comparatively, the standard deviation of our method is steady thanks to the update process. Note how the spikes of blue curve in figure (72)(b) are distinguishable and therefore detectable.

4.8.2. *Test Time/DPPM Comparison.* So far, we have discussed the success of the proposed method in identifying outliers. But how is it useful in production line? First of all, it can be used to eliminate outlier devices to guarantee reliable operation in application areas where reliability is important. Another application area of the method is attaching it to existing test framework in production line to boost test quality. In this work, we integrate the proposed method to an adaptive test framework in order to improve defect coverage.

In this work we demonstrate adaptive multidimensional outlier analysis in a test optimization framework, where the number of applied tests is reduced by skipping some of them on-the-fly by the test engine. Therefore, measurement data analyzed by our method includes only a subset all specification parameters and measured parameters differ from device to device.

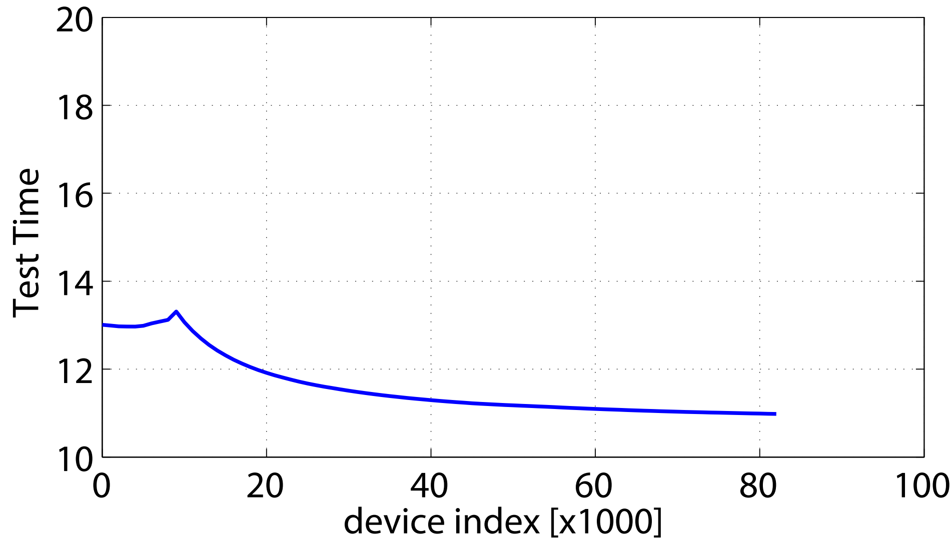


Fig. 73. Test time with respect to testing progress for the second data set

We show averaged test time with respect to the progress in testing in Figure 73. Note that test time does not increase as testing progresses and is stable.

Table 14 compares DPPM/Test Time performance for the first circuit. Test times are given in terms of the average number of tests executed. Achieved DPPM level is lower compared to the adaptive method [Yilmaz and Ozev(2010)] and significantly lower compared to the other static methods. Note that integrating our method into [Yilmaz and Ozev(2010)] yields even lower DPPM while reducing the required test time almost to half.

Table 15 shows performance results for the 2nd circuit. The achieved DPPM result is one third of the best of the other methods for comparable test time. Embedding our method into [Yilmaz and Ozev(2010)] resulted only %5 increase in test time, but this enabled to reduce DPPM by 3 folds.

In Table 16, we show the improvement of test quality when the proposed method is used for the 3rd data set. Test engine is set to minimize DPPM level. Test set is reduced from

	DPPM	Time
Proposed	36	11
[Yilmaz and Ozev(2010)]	43	20
Cover Based	392	20
ILP [Stratigopoulos *et al.*(2007)]	800	5
Marginality Based [Chen and Orailoglu(2008)]	135	15.5
Heuristic [Milor(1998)]	590	14.34
static SVM [Biswas *et al.*(2005)]	735	17.4

Table 14. DPPM and test time results for data set-1

	DPPM	Time
Proposed	47	7.33
[Yilmaz and Ozev(2010)]	164	7
Cover Based	1.4k	7
ILP [Stratigopoulos *et al.*(2007)]	1k	3.5
Marginality Based [Chen and Orailoglu(2008)]	1.9k	3
Heuristic [Milor(1998)]	150	6.56
static SVM [Biswas *et al.*(2005)]	664	17.2

Table 15. DPPM and test time results for data set-2

264 tests to 80 tests, whereas DPPM is nearly 100. Compared to static methods, adaptive test and outlier analysis have a significant advantage in terms of DPPM. Outlier analysis

	DPPM	Time
Proposed	105	80
[Yilmaz and Ozev(2010)]	107	81.2
Cover Based	621	82
ILP [Stratigopoulos *et al.*(2007)]	3.8k	9.2
Marginality Based [Chen and Orailoglu(2008)]	4.3k	10.9
Heuristic [Milor(1998)]	1.87k	142
static SVM [Biswas *et al.*(2005)]	1.56k	47.6

Table 16. DPPM and test time results for data set-3

does not provide a significant improvement over our previous work for this case. This can be attributed to the lack of correlation of the outliers to defective instances.

Results show that integrating the proposed method into an existing test engine can yield significant improvements.

4.9. *Summary*

This work presents an adaptive multidimensional outlier analysis technique that enables extraction of information from multi-variate measurement parameter space and selects only information containing parameters through uncertainty aware selection. Uncertainty aware optimized test selection gives rise to high DUM (Detection Unit Metric) and therefore high detection probability. Moreover, continuously updating characterization information enables us to track process shifts and adapt with respect to them.

Flexibility of the method makes it suitable for industry applications, since it can be integrated to existing test frameworks to achieve very low DPPM levels while reducing the

test time considerably. In fact both test time and test quality can be improved significantly simultaneously as we have observed in the results.

5. Efficient Process Shift Detection and Test Re-Alignment

Efficiency of test compaction is very important for production test time minimization. Poor test compaction methods either result in long test time or low test quality for analog and mixed-signal circuits. One of the most important factors in test compaction quality is accuracy of the representation of process statistics. Accurate representation is challenging since process characteristics are not stationary; thus, they need to be updated to maintain a reliable test quality level over the complete production run. Previous work in test compaction either does not take process shift into account or uses simplistic updating methods to avoid the cost of process re-learning.

In this work, we propose an efficient re-learning method that tracks changes in the process state and generates a compact test list using re-learned information. We model the mechanics of the process shift with a transformation function. We use information from a characterized wafer to predict the characteristics of a given wafer under test (WUT) using a very small number of samples. Fitting the transformation function enables us to map outdated process information to the up-to-date process information of the WUT. We demonstrate the performance our method and compare it with previously published work using large scale production data of two distinct mixed-signal circuits. We show that our method maintains superior DPPM levels over large numbers of wafers and lots.

5.1. *Motivation*

Process characteristics of manufactured devices change over time. These changes may be due to intentional adjustments to tweak the process or due to difficulties in providing precisely controlled production steps.

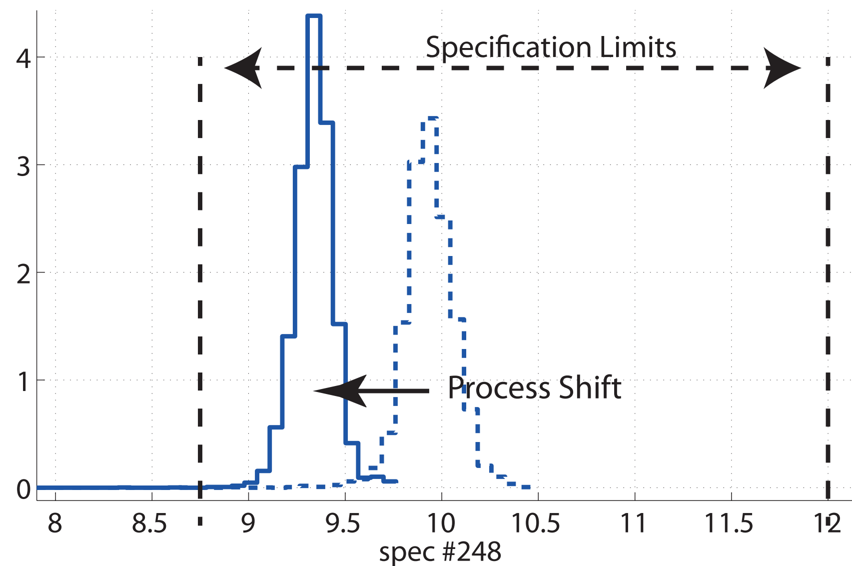


Fig. 74. Process shift makes specification parameter\#248 move toward the lower specification limit, increasing probability of failure.

We illustrate this scenario using production data of a large scale analog circuit. Figure 74 shows the probability distribution function (PDF) of a specification parameter (spec# 248) generated using information collected from two separate wafers from the same lot. The dashed curve shows the PDF of the parameter from the first wafer, while the solid curve shows the PDF of the same parameter from the second wafer. The figure shows that PDF of the same parameter shifts towards the lower (left) specification boundary. Initially, this parameter has a negligible probability of failure and is not included in the compacted test list. However, the shift makes it a critical parameter and necessitates it to be included in production test list to avoid potential misclassifications. This example shows that a parameter that is initially not critical becomes critical after a wafer transition. Such information would be missed in a lot-to-lot based update frequency. Thus, a better tracking and update approach is needed to maintain high test quality.

Updating characterization data is particularly challenging because capturing specification-to-specification dependencies for effective test compaction requires a significant re-learning effort, typically requiring full test of thousands of devices. Throwing away outdated information and collecting new data from scratch is not the most efficient way of re-learning a process. Considering that the process technology remains the same, shifted process statistics cannot be independent of the outdated (available) information. Process re-learning can be performed in a much more efficient way if state-to-state process correlations are used.

In the following analysis, we illustrate that wafer-to-wafer changes do not require a full re-learning step. In (3.26), (3.27), and (3.28), we define a well known metric, CPK, difference CPK (DCPK) and a utility metric in terms of the CPK metric.

$$CPK_k{:}W_i = min\left(\frac{(\mu_k - spec^-)}{3\sigma_k}, \frac{(spec^+ - \mu_k)}{3\sigma_k}\right) \quad (3.26)$$

$$DCPK_k : W_i = \frac{|CPK_k : w_i - CPK_k : w_0|}{CPK_k : w_0} \quad (3.27)$$

$$CM_i[\%] = 100 \cdot \sum_k DCPK_k : w_i \quad (3.28)$$

where μ_k and σ_k are the mean and the standard deviation values of the kth parameter on wafer number i. Equation (3.28) normalizes the changes in CPKs and (3.28) sums them to yield an overall measure for all parameters on the same wafer. Normalization enables us to assign all CPKs a common meaning and to compare them with each other. Normalization equation (3.27) emphasizes the critical specifications that may lead to a wrong decision in production test. A DCPK value close to 0 indicates that even if there is a shift, it has negligible effect on the decision mechanism. On the other hand, a DCPK value approaching

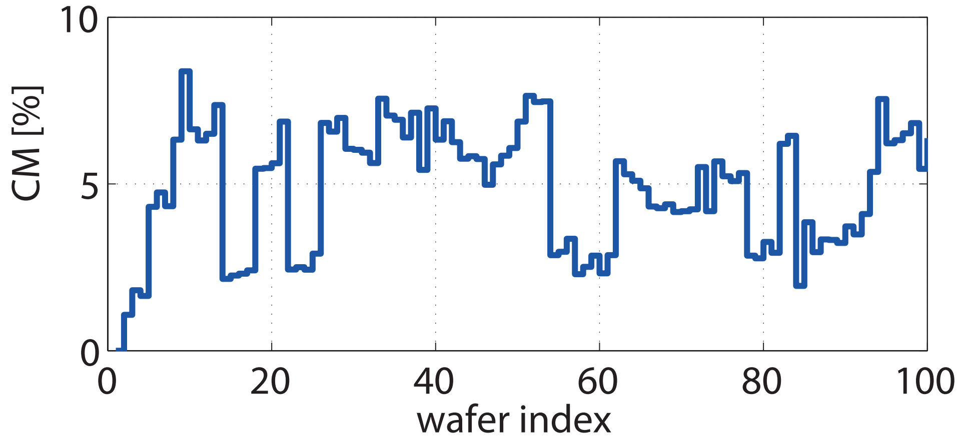


Fig. 75. Wafer-to-wafer process statistics are typically correlated. For most transitions there is only a little change

to 1 indicates a significant change that needs to be taken into account in the test compaction phase.

Parameter CM (given in Equation (3.28)) represents the overall change of the process state in terms of $DCPK$ values. As an example, the CM values of a production data set are plotted for the first 100 wafers in Figure 75 across several lots. The figure shows that most of the time there is little change in CM, indicating that wafer-to-wafer information can be highly correlated.

The main challenge of efficient re-learning is to determine how frequent re-learning should be performed and what is the most efficient re-learning strategy. In [Benner and Boroffice(2001)], the authors propose a lot-to-lot re-learning. However, this strategy does not guarantee to capture potential shifts within a lot and is dangerous as it may lead to misclassifications. Note that Figure 75 shows that within-lot variation can also be significant. Therefore, the update period should be short enough to capture these changes. In [Gotkhindikar *et al.*(2011)], a continuous update and test ordering mechanism is adopted

to monitor pass/fail statistics of the executed tests. However, this method has a limited test compaction potential and achieves good results for low yield processes. We re-learn at all wafers to avoid test quality degradation and collect sufficient data to capture specification dependencies to achieve a high compaction rate.

5.2. *Methodology*

Accuracy of the characterization data is of crucial importance for reliable and high quality test compaction since a large training data set is required to capture less probable failure patterns. Full re-characterization is performed in previous work [Benner and Boroffice(2001)] to maintain a reliable and up-to-date characterization data. However, this strategy is not efficient because the correlation between the process states is ignored. We use wafer-to-wafer correlations to capture this information and to minimize the re-characterization overhead.

In this section, we concentrate on specification parameter domain process tracking. Process state can be represented as a stochastic process with spatial and temporal dependencies, $\Gamma_i(x, y)$, where the temporal dependency corresponds to wafer to wafer correlation. Within this representation, i is the wafer index, while x and y are coordinate parameters on the wafer. Equation (3.29) shows the formulation of process shift with this statistical framework.

$$\Gamma_0(x, y) \quad \overset{T_{0,i}}{\longrightarrow} \quad \Gamma_i(x, y) \tag{3.29}$$

where $T_{0,i}$ is a function that represents the state transition of the process. Even if the process Γ_0 is fully characterized, it shifts to Γ_i through $T_{0,i}$. Instead of re-characterizing the new

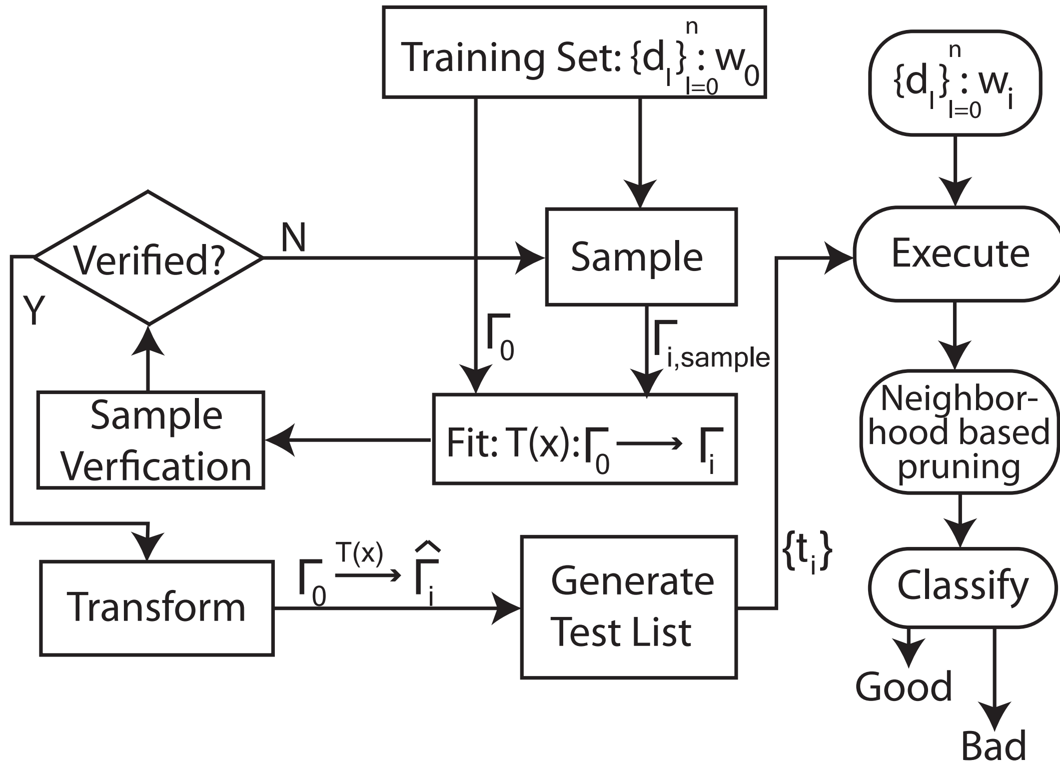


Fig. 76. Process state is re-learned at every wafer transition. The new wafer is sampled until a consistent transformation relation (T) is fitted between old process state and the new process state. Then, test list is generated using updated process statistics.

process state, we model the transition function ($T_{0,i}$) and map the available characterization data (Γ_0) to obtain the up-to-date process state (Γ_i) using the generated model.

We model the transition function using an affine transformation defined in multiple sub-regions of the wafer. Modeling using affine transformations enable us to use a very small number of samples for model fitting. Dividing the wafer into sub regions and fitting a transition function for each region enables us to approximate arbitrarily complex transition relationships.

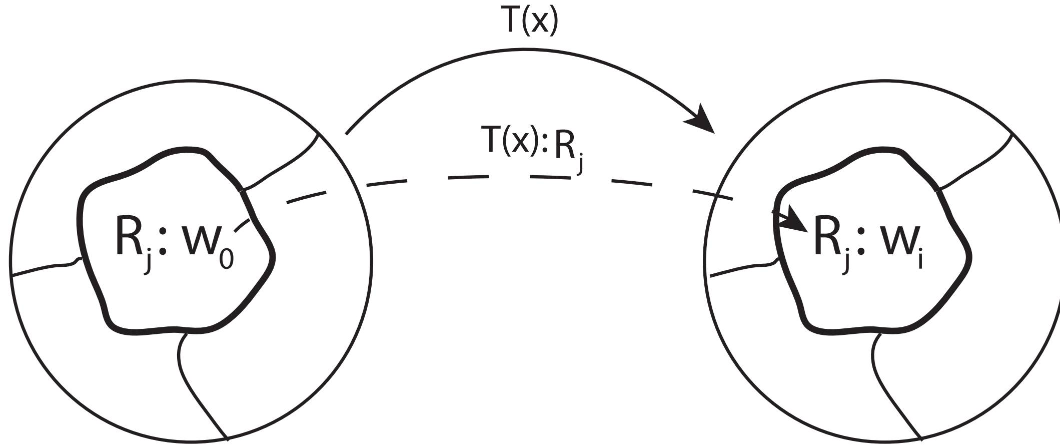


Fig. 77. Wafers are virtually divided into sub-regions and transformation functions are fitted to relate the devices in these sub-region.

Once an up-to-date process state is generated, we update the test list in order to classify devices correctly. Test list is re-generated for each wafer to achieve wafer level test compaction adaptation. Note that once a test list is determined, it is applied to all devices until the next update step. This enables us to use multi-site testing easily to achieve further test time reduction. Also note that this data collection and update process can be applied to multiple wafers in parallel.

5.3. *Modeling Approach*

The transition function that we try to model can be a complex and nonlinear function in general. However, we simplify the model by linearizing it using a power series expansion around multiple spatial locations on the wafer. This enables us to model functions with arbitrary precision as long as the wafer is sufficiently partitioned. Thus, the complexity of the transition function is reduced. Power series expansion equation for our model is given in equation (3.30).

$$\Gamma_i(x_0, y_0) = \alpha_0 + \alpha_1\Gamma_0(x_0, y_0) + \alpha_2\Gamma_0^2(x_0, y_0) + \cdots \tag{3.30}$$

where the transition function is linearized around (x_0, y_0) on the wafer. Note that we only use the first two terms for modeling. Therefore, we only need to estimate α_0 and α_1. We estimate these coefficients by testing a small number of samples form W_i. The update/re-learning procedure can be formulated as estimating the statistics of Γ_i (Process statistics of the ith wafer) using minimum number of samples while maintaining a certain fitting quality level. We thus effectively transform the problem of fitting a complex function ($T_{0,i}$) to the problem of fitting the smallest linear transformation function set possible.

We attack this problem in two steps. We first divide the characterization wafer and wafer under test (WUT) in sub-regions. Then, we generate a transition model from W_0 to W_i by sampling a few initial points. We verify the generated model using another small set of samples. The regions that are not modeled adequately are divided into sub regions and this modeling/verification procedure is applied until all sub-regions are modeled with an acceptable accuracy. The flow of the model fitting procedure is shown in Figure 76.

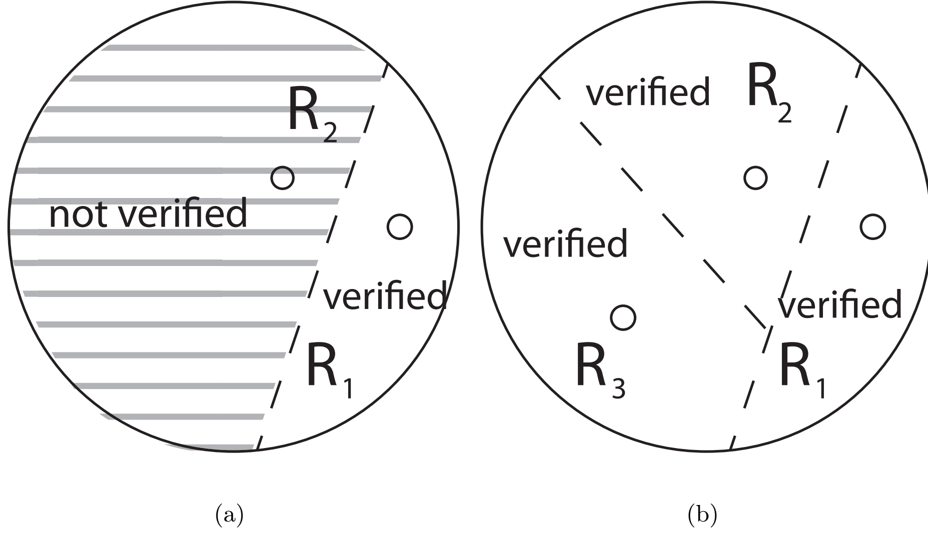


Fig. 78. Wafers are partitioned and a transformation function is fitted for each partition separately. Sub-partitions are re-partitioned to approximate complex transformation functions. Two partitions are used in (a) and transformation function of partition R_2 is not verified (shaded). R_2 is partitioned into two in (b) to achieve a better approximation.

We start partitioning by selecting two random devices on the characterization wafer, which we call pole devices, and group the other devices on the wafer into two groups based on the spatial distance from these pole devices using equation is given in (3.32).

$$D_{l,j} = \left\| d_l^{XY} - d_j^{XY} \right\| \tag{3.31}$$

$$d_l : R_j \leftarrow \underset{j}{argmax} \ \frac{1}{D_{l,j}} > \frac{1}{D_{l,k}}, k \neq j \tag{3.32}$$

where the lth device (d_l) on the wafer is assigned to j^{th} region if it is spatially closest to the jth pole according to L_2 distance metric. d^{XY} indicates device location on the WUT.

Once the partitions are selected, several additional devices are characterized on the WUT in all regions. These samples are used to generate a linear transformation from $W_0 : R_1$ to $W_i : R_1$ and $W_0 : R_2$ to $W_i : R_2$ using least squares regression approach. At least two samples are required from each region to be able fit a linear transformation equation. We take an oversampling approach to reduce the error due to the measurement errors and non-systematic process variation. We use 5 samples instead of 2.

Then, we select another 5 samples from the WUT and generate a model for verification. If the generated models produce statistically similar estimations, the generated model is used (verification succeeds). Otherwise, the region is subdivided into two sub-regions and model generation/verification process is applied until all sub-regions are modeled with an acceptable accuracy.

If the wafer-to-wafer transition is linear (or sufficiently linear) in all specification parameters, the model is generated using as low as 20 samples. However, if at least one of the parameters have more complex transformation relationship, more samples are required. Partitioning enables to narrow the neighborhood of localization in the parameter space.

5.4. *Verification*

Model verification is performed in the specification parameter domain. We use the generated transformation model $T_{0,i}$ and verification model $T_{0,i}^V$ to estimate device performances on the WUT using equations:

$$
\begin{aligned}
\hat{s}_{l,k} : W_i &= T_{0,i,k}(s_{l,k} : W_0) \\
\hat{v}_{l,k} : W_i &= T_{0,i,k}^V(s_{l,k} : W_0)
\end{aligned}
$$

where, $s_{l,k} : W_i$ represents k^{th} specification of lth device on the ith wafer, $\hat{s}$ and $\hat{v}$ represent generated estimation and verification parameters. We test whether $\hat{s}$ and $\hat{v}$ are statistically different from each using Kolmogorov-Smirnoff (KS) test. The KS test checks if the hypothesis of the distributions of the to estimations are same is correct or not. We use 95% significance level to reject the hypothesis test.

Note that $\hat{s}$ and $\hat{v}$ are not assumed to have any parametric distribution. Neither the transformation model nor the hypothesis testing imposes any restrictions the representation of the specification mode. This is very important to be able to capture complex and nonparametric analog parameters.

5.5. *Neighborhood Based Pruning*

Devices that pass all the applied test are identified as potentially good devices and are subjected to a sanity check step before the final classification. The available measurement information is used to perform a final check in order to minimize the risk of a potential misclassification. We use neighborhood information to identify suspicious devices by using spatial dependency of process variation to reduce the uncertainty level.

In the pruning step, devices are tested with respect to the statistics of the process using a 6σ window. Devices that reside inside this window are accepted, while the others are subjected to full test. We use local information to generate a 6σ window for each test by computing the window by using spatially nearest 8 devices. Neighbor devices experience smaller variation and enable to achieve a much narrower acceptance window and therefore a finer filtering mechanism due spatial dependency of process variation.

5.6. *Test Compaction*

The proposed process shift tracking methodology is a general purpose method and can be integrated into any test engine. We demonstrate the update mechanism by integrating it into the well known set-cover test method for the simplicity of analysis. However, we use a modified version of the set-cover method that is resilient to misrepresentation of process statistics due to the limited size of the characterization data set.

Set-cover problem is formulated in Eq (3.33).

$$\begin{aligned} Goal: \quad & t_{min} = min \quad |t| \\ Constraint: \quad & A_{i,j}t \geqslant \quad \underline{1} \end{aligned} \tag{3.33}$$

where, A[1] is the fallout matrix of size $n \times m$ (n is the number of failing devices and m is the total number of tests), t is test vector, $\underline{1}$ is a unity vector same size as t, and $|.|$ is cardinality function. Entries of $A_{i,j}$ matrix is set to 1 if i^{th} device fails j^{th} test and 0 if otherwise. The goal of the problem is to find a minimum size test list t_{min} that identifies all failing devices in the training set. The accuracy of the set-cover method depends on the representativeness of matrix A. Large and up-to-date data required to obtain a representative A matrix. This makes the set-cover method sensitive to training set size which is typically limited due to high characterization cost.

The set-cover method uses characterization data as if matrix A represents fail patterns perfectly. However, fail patterns are probabilistic due to process variation and various uncertainties in measurements. Matrix A is a realization of the probabilistic system with a limited sample size. Failing to recognize this probabilistic nature in test compaction may lead to unwanted test escapes.

[1] $A_{i,j}$ is a binary matrix that contains pass/fail information of jth test of the ith device.

This concept can be explained through a simple example. Suppose that test#1 is ideally 90% correlated with test#2, so test#1 statistically covers test#2 90% of the time. However, according to a particular training data set, all devices that fail test#1 also fail test#2 (100% correlation is estimated) and set-cover method drops test#2 due to its perfect correlation with test#1. However, they are not perfectly correlated and this wrong judgment may result in %10 escape rate since test#2 is not applied.

In order to mitigate the effect of such errors, we use 2-detect approach in test selection. This helps to generate a robust test list by selecting backup tests to cover the fail matrix. Therefore the constraint of our method becomes:

$$Constraint: \quad A_{i,j}t \geqslant \quad \underline{2}$$

5.7. *Results*

We apply our proposed method to production data of a large-scale analog circuit and a mixed signal circuit. The first data set consists of exhaustive measurement results of 264 specification parameters of ~900k devices, while the second data set has 365 measurement results of 1600k devices. The data is collected across tens of lots and hundreds of wafers. First, we discuss the process transformation concept we introduced in section 5.3 and demonstrate it using production data. Then, we integrate the re-learning mechanism with the set-cover based test selection method presented in section 5.6 to show the performance of our proposed method.

5.7.1. *Modeling Results.* Figure 79 illustrates the performance of the proposed process state transformation method. In Figure 79(a), measurement results of devices from two different wafers are scatter plotted to show wafer-to-wafer dependencies. Horizontal axis show

the performance of the devices on wafer#0, while the vertical axis show the performance on wafer#1. Histograms plotted at the side and the bottom of the figure show PDFs of the parameter. The dashed diagonal line is an ideal line along which devices should align when there is no shift. The geometric location of the devices indicate that device performances of two different wafers are correlated, while the correlation is not perfect. This observation supports our claim on the state-to-state correlation of the process states. As expected, there is a dependency that enables us to make prediction whereas there is a certain amount of uncertainty that imposes a lower limit on the accuracy we can achieve. Note that the plotted device ensemble does not overlap with the dashed ideal line, indicating a process shift. The bulk of the devices reside above the dashed line. The effect of the shift is also observable in the PDFs plotted along the horizontal and vertical axis.

We apply our modeling approach to model the transformation from wafer#0 to wafer#1. The model is generated and verified by using 30 samples and 3 regions. Then, device set in wafer#0 is transformed through the fitted function. We show the goodness of the estimate in Figure 79(b) by a scatter plot of the estimated results with respect to the actual parameters of wafer#1. Note that the devices are located around the 45 degree dashed line indicating a good fit. The residual error arises mainly from the random variation that cannot be predicted. An important observation here is that the absolute error mode in the prediction is less important than having the samples equally spaced around the 45 degree line since that impacts the distribution statistics. The PDF juxtaposition on the samples show significant improvement from (a) to (b). This example shows that a small number of devices can be used to predict systematic dependencies among wafers and predict statistics of the wafer of interest.

5.7.2. *Production Data.* Process re-learning is performed after every wafer transition to minimize the risk of missing fast process changes. Therefore, the re-learning process requires a certain amount of time that should be at a reasonable level. In Figure 80, we show the number of samples used for the proposed re-learning method for the first data set for the first 100 wafers. Horizontal axis shows wafer indexes and vertical axis shows the number of samples including the verification samples. For most of the wafers, a sample size as small as 20 is sufficient, while a few wafers require around 200 samples. On the average, 40 samples are used in the re-learning process.

We integrate the proposed re-learning algorithm with the set-cover based method described in section 5.6 and report DPPM and test time performance in Figure 81. Reported DPPM is calculated by projecting the number of defect escapes to represent parts per million, while time reduction is calculated by averaging the number of eliminated tests and dividing by the overall number of tests. A cover-based method is implemented to serve as the baseline for comparison purposes. Test list for the cover-based method is ordered off-line (static) with respect to the coverage rate of the tests and trimmed to obtain different size test lists. We used these different sized test lists to reveal the trade-off on the DPPM-test time plot. The family of points obtained using the cover-based method are marked using rectangle symbols on the plot and connected by interpolation. The goal of test compaction methods is to achieve a small test time and low DPPM level. We compare the performance of some of the key works in this field using this baseline performance plot and the proposed method.

Test list generated using a continuous process shift adaptation method [Gotkhindikar *et al.*(2011)] is trimmed using a drop rate factor to achieve a family of test lists and therefore

a set of trade-off points on the plot. This method achieves good test compression for low yield processes and therefore not very convenient for high yield processes.

The set-cover method does not adapt to process shifts and uses the initial training set of devices for test compaction. The set-cover method yields a lower test time; however, DPPM level is significantly high. The results of lot-to-lot learning [Benner and Boroffice(2001)] method also show high DPPM level, comparable to the cover based method. Our method achieves more than ten-fold DPPM reduction over the static set-cover method and is significantly below any of the trade-off curves generated.

The improvement can be attributed to re-learning method that enables to generate a test list adaptively for each wafer and pruning check step.

We also compare the proposed method with an adaptive test method [Yilmaz *et al.*(2011)]. This method achieves the lowest DPPM level for all data sets due to its per-device adaptation method. It can be applied for tester platforms that support device specific test tailoring, which is typically supported by high end platforms. Our method imposes much less burden on the tester and can be implemented on low end testers. Moreover, since our method applies a fixed test list to all devices on the WUT, it can be easily adopted to multi-site test which enables linear test time reduction with respect to the number of sites.

Table 17 summarizes the differences of the proposed method, [Yilmaz and Ozev(2010)], set-cover based method, and the per-device adaptive method in [Yilmaz *et al.*(2011)]. Executing the same test order for all devices enables input sharing. Test time reduction rate for multi-site test is linear with respect to the number of parallel sites. The computational burden of the proposed method is negligible. On the average, processing time of the relearning step is less than 2ms per device on a quad-core 2.3GHz Intel machine. This time

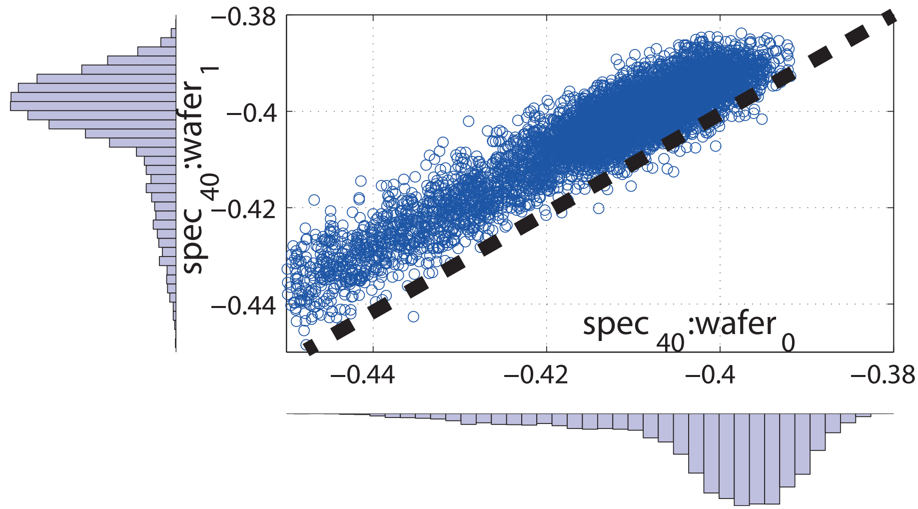
	proposed	[Yilmaz and Ozev(2010)]	set-cover
Requires Different Test Inputs	no	yes	no
Static Compaction	yes	no	yes
Process shift adaptation	very good	acceptable	not available
Test reduction factor for M-site test	M	<M	M
Computational burden	negligible	modest	no-burden

Table 17. Summary of the main features

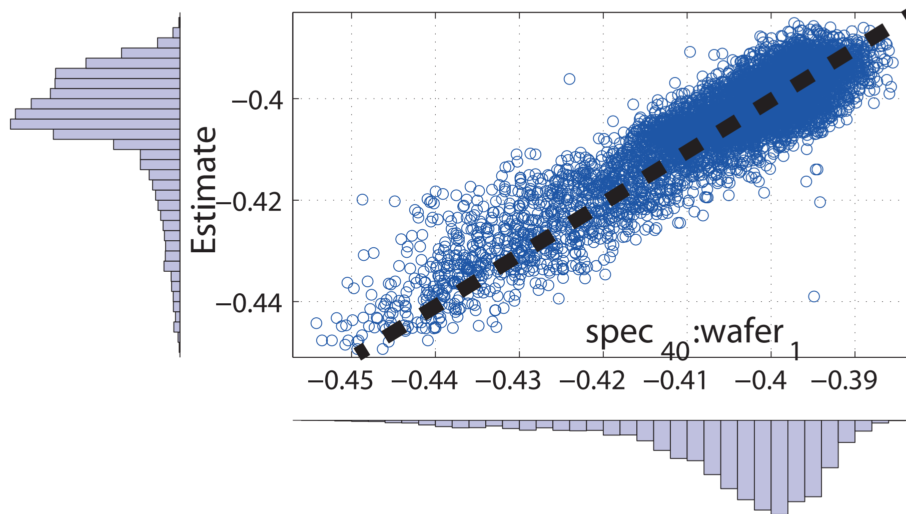
is negligible compared to the average test time of a device, which is typically in the order of seconds.

5.8. *Summary*

We present an efficient process state tracking method that enables us to update process information during production test using a very small number of samples. Using the updated information, we can select a compact test list tailored with respect to the statistics of the wafer under test. As a result, we achieve a reliable and consistent test quality level over the production life-time. We model a transformation function relating a known state of a process to an unknown state using an affine transformation defined in sub-regions of the wafer. A simple transformation model enables us to generate a fitting function using a small number of samples, while sub-dividing the wafer into regions enables us to approximate the transformation functions of arbitrary complexity. We integrated the proposed process tracking method with a set cover based method and a neighborhood based pruning step to show its potential in product test environment. The results show that our method achieves several folds improvement in test time/DPPM trade-off over the previous work.



(a)



(b)

Fig. 79. Scatter-plot of the same parameter of two different wafers shows a systematic shift in (a). The proposed modeling approach successfully approximates the distribution of the parameter of the new wafer. Scatter-plot and ideal 45 degree line overlap in (b).

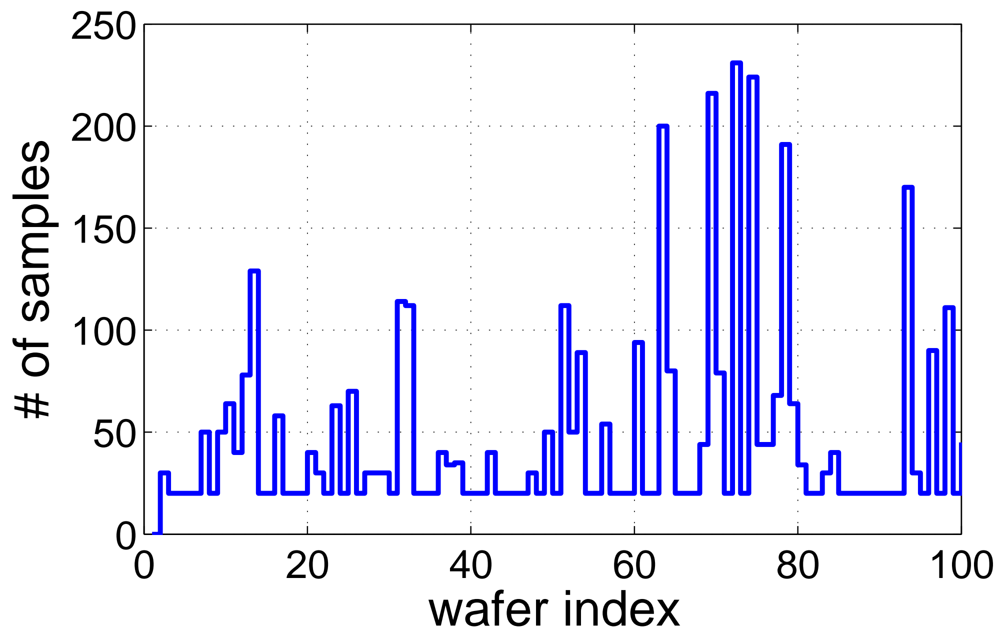
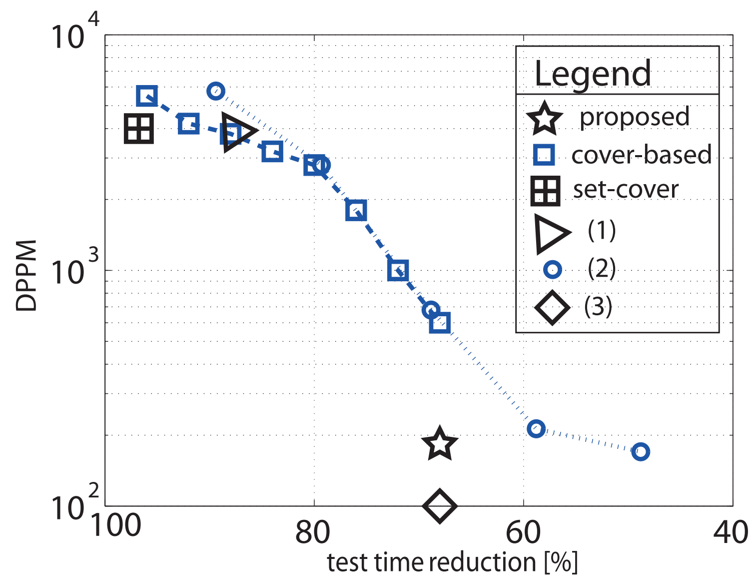
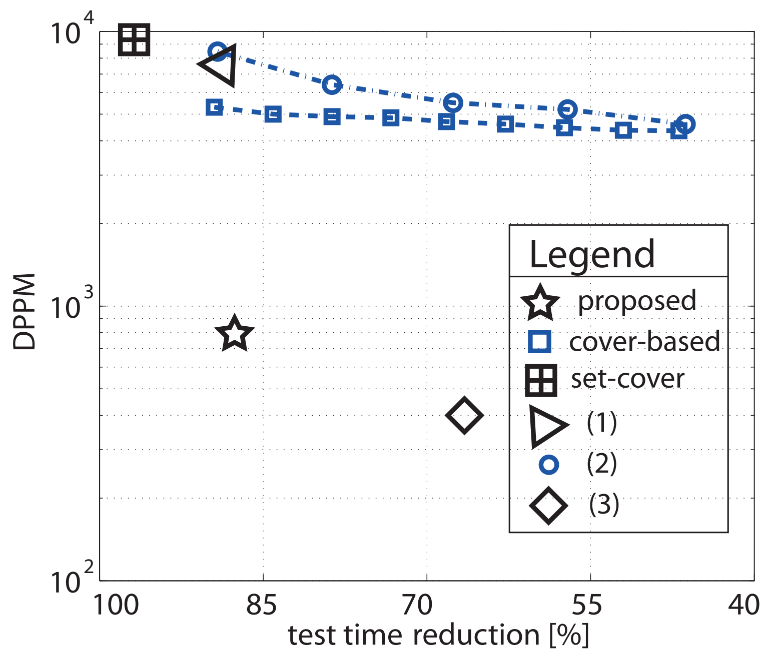


Fig. 80. Most of the wafers require as low as 20 samples for re-learning. The average re-learning cost is 40 samples per wafer.



(a)



(b)

(1) [Bhattacharya *et al.*(2005a)], (2) [Bhattacharya *et al.*(2005a)], (3) [Yilmaz *et al.*(2011)]

Fig. 81. Production data results for (a) data set-1, (b) data set-2.

REFERENCES

(1999). IEEE std. 802.11.a-1999.

(2010). Neural network toolbox. Mathworks.

Acar, E. and Ozev, S. (2008). Defect-oriented testing of RF circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **27**(5), 920 –931.

Acar, E., Ozev, S., and Redmond, K. (2006). Enhanced error vector magnitude EVM measurements for testing WLAN transceivers. *Computer-Aided Design, 2006. ICCAD '06. IEEE/ACM International Conference on*, pages 210–216.

Acar, E., Ozev, S., Srinivasan, G., and Taenzler, F. (2008). Optimized EVM testing for IEEE 802.11a/n RF ICs. *Test Conference, 2008. ITC 2008. IEEE International*, pages 1–10.

Akbay, S. and Chatterjee, A. (2004). Feature extraction based built-in alternate test of RF components using a noise reference. In *IEEE VLSI Test Symposium*, pages 273–278.

Akbay, S. and Chatterjee, A. (2007). Fault-based alternate test of RF components. In *IEEE International Conference on Computer Design*, pages 518 –525.

Azais, F., Bertrand, Y., Renovell, M., Ivanov, A., and Tabatabaei, S. (2003). An all-digital DFT scheme for testing catastrophic faults in PLLs. *IEEE Design Test of Computers*, **20**(1), 60 – 67.

Benner, S. and Boroffice, O. (2001). Optimal production test times through adaptive test programming. In *IEEE International Test Conference*, pages 908–915.

Bhattacharya, S., Halder, A., Srinivasan, G., and Chatterjee, A. (2005a). Alternate testing of RF transceivers using optimized test stimulus for accurate prediction of system specifications. *Journal of Electronic Testing*, **21**(3), 323–339.

Bhattacharya, S., Senguttuvan, R., and Chatterjee, A. (2005b). Production test enhancement techniques for MB-OFDM ultra-wide band (UWB) devices: EVM and CCDF. *Test Conference, 2005. Proceedings. ITC 2005. IEEE International*, pages 10 pp.–245.

Bishop, A. and Ivanov, A. (1995). Fault simulation and testing of an OTA biquadratic filter. In *IEEE International Symposium on Circuits and Systems*, volume 3, pages 1764 –1767.

Biswas, S. and Blanton, R. D. S. (2006). Statistical test compaction using binary decision trees. *IEEE Design and Test of Computers*, **23**(6), 452–462.

Biswas, S. and Blanton, R. S. (2008). Test compaction for mixed-signal circuits using pass-fail test data. *IEEE VLSI Test Symposium*, pages 299–308.

Biswas, S., Li, P., Blanton, R., and Pileggi, L. (2005). Specification test compaction for analog circuits and MEMS [accelerometer and opamp examples]. *IEEE DATE*, **1**, 164–169.

Cabral, R., Escarigo, S., Neto, H., and Sarmento, H. (2006). Implementation of a DAB receiver with FPGA technology. *Consumer Electronics, 2006. ICCE '06. 2006 Digest of Technical Papers. International Conference on*, pages 397–398.

Cerioli, A. (2009). Finding an unknown number of multivariate outliers. *Journal of the Royal Statistical Society*, **71**(2), 447–466.

Chang, S.-J., Lee, C. L., and Chen, J. E. (2002). Structural fault based specification reduction for testing analog circuits. *J. Electron. Test.*, **18**(6), 571–581.

Chao, C.-Y., Lin, H.-J., and Miler, L. (1997). Optimal testing of VLSI analog circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **16**(1), 58 –77.

Chen, M. and Orailoglu, A. (2008). Test cost minimization through adaptive test development. In *IEEE International Conference on Circuit Design*, pages 234–239.

Cormen, T. H., Leiserson, C. E., and Rivest, R. L. (1997). *Introduction to Algorithms*. McGraw-Hill.

Daasch, R., McNames, J., Bockelman, D., Cota, K., and Madge, R. (2000). Variance reduction using wafer patterns in IDDQ data. In *IEEE International Test Conference*, pages 189–198.

Daasch, R., Cota, K., McNames, J., and Madge, R. (2001). Neighbor selection for variance reduction in IDDQ and other parametric data. In *IEEE International Test Conference*, pages 92–100.

Daasch, R., Cota, K., McNames, J., and Madge, R. (2004). In search of the optimum test set. In *IEEE International Test Conference*, pages 203–212.

D'Errico, J. R. and Zaino, Nicholas A., J. (1988). Statistical tolerancing using a modification of taguchi's method. *Technometrics*, **30**(4), 397–405.

Drineas, P. and Makris, Y. (2003). Independent test sequence compaction through integer programming. *IEEE International Conference on Computer Design*, page 380.

Fang, L., Lemnawar, M., and Xing, Y. (2006). Cost effective outliers screening with moving limits and correlation testing for analogue ics. *IEEE Test Conference*, pages 1–10.

Filzmoser, P., Garrett, R. G., and Reimann, C. (2005). Multivariate outlier detection in exploration geochemistry. *Computers and Geosciences*, **31**(5), 579 – 587.

Gentle, J. E. (2003). *Random Number Generation and Monte Carlo Methods*. Springer.

Gotkhindikar, K., Daasch, W., Butler, K., Carulli, J., and Nahar, A. (2011). Die-level adaptive test: Real-time test reordering and elimination. In *IEEE Internationa Test Conference*, pages 1 –10.

Gu, Q. (2005). *RF System Design of Transceivers for Wireless Communications*. Springer Science.

Hafed, M. and Roberts, G. (2000). A stand-alone integrated excitation/extraction system for analog BIST applications. In *IEEE CICC*, pages 83 –86.

Haider, A. and Chatterjee, A. (2005). Low-cost alternate EVM test for wireless receiver systems. *VLSI Test Symposium, 2005. Proceedings. 23rd IEEE*, pages 255–260.

Halder, A., Bhattacharya, S., and Chatterjee, A. (2008). System-level specification testing of wireless transceivers. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, pages 263–276.

Hanzo, L., Munster, M., Choi, B., , and Keller, T. (2003). *OFDM and MC-CDMA for Broadband Multi-User Communications, WLANs and Broadcasting*. Wiley-IEEE Press.

Helfenstein, M., Baykal, E., Muller, K., and Lampe, A. (2005). Error vector magnitude EVM measurements for GSM/EDGE applications revised under production conditions. *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pages 5003–5006 Vol. 5.

Huss, S. and Gyurcsik, R. (1991). Optimal ordering of analog integrated circuit tests to minimize test time. In *IEEE/ACM Design Automation Conference*, pages 494–499.

Jiang, M. F., Tseng, S. S., and Su, C. M. (2001). Two-phase clustering process for outliers detection. *Pattern Recognition Letters*, **22**(6-7), 691 – 700.

Jiang, W. and Vinnakota, B. (1999). Defect-oriented test scheduling. *IEEE VLSI Test Symposium*, pages 433–438.

Kanj, R., Joshi, R., and Nassif, S. (2006). Mixture importance sampling and its application to the analysis of SRAM designs in the presence of rare failure events. In *IEEE DAC*, pages 69–72.

Kupp, N., Drineas, P., Slamani, M., and Makris, Y. (2009). On boosting the accuracy of non-RF to RF correlation-based specification test compaction. *Springer Journal of Electronic Testing*, **25**, 309–321.

Kupp, N., Slamani, M., and Makris, Y. (2011). Correlating inline data with final test outcomes in analog/RF devices. In *IEEE Design, Automation Test in Europe Conference Exhibition, 2011*, pages 1–6.

Liu, C.-H. (2003). On the design of OFDM signal detection algorithms for hardware implementation. *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE*, **2**, 596–599.

Madge, R., Macchetto, C., Rajagopalan, V., B.H.Goh, Daasch, R., and Schuemeyer, C. (2002a). Screening minVDD outliers using feedforward voltage testing. In *IEEE VLSI Test Symposium*, pages 673–682.

Madge, R., Rehani, M., Cota, K., and Daasch, W. (2002b). Statistical post-processing at wafersort-an alternative to burn-in and a manufacturable solution to test limit setting for sub-micron technologies. In *IEEE VLSI Test Symposium*, pages 69 – 74.

Maeda, T., Matsuno, N., Hori, S., Yamase, T., Tokairin, T., Yanagisawa, K., Yano, H., Walkington, R., Numata, K., Yoshida, N., Takahashi, Y., and Hida, H. (2006). A low-power dual-band triple-mode WLAN CMOS transceiver. *Solid-State Circuits, IEEE Journal of*, pages 2481–2490.

Mak, T., Tripp, M., and Meixner, A. (2004). Testing gbps interfaces without a gigahertz tester. *IEEE Design Test of Computers*, **21**(4), 278 – 286.

Maxwell, P., O'Neill, P., Aitken, R., Dudley, R., Jaarsma, N., Quach, M., and Wiseman, D. (2000). Current ratios: a self-scaling technique for production IDDQ testing. *IEEE International Test Conference*, pages 1148–1156.

Meixner, A. and Maly, W. (1991). Fault modeling for the testing of mixed integrated circuits. In *IEEE International Test Conferencel*, page 564.

Meixner, A., Kakizawa, A., Provost, B., and Bedwani, S. (2008). External loopback testing experiences with high speed serial interfaces. In *IEEE International Test Conference*, pages 1 –10.

Menolfi, C., Toifl, T., Buchmann, P., Kossel, M., Morf, T., Weiss, J., and Schmatz, M. (2007). A 16gb/s source-series terminated transmitter in 65nm CMOS SOI. In *IEEE International Solid-State Circuits Conference*, pages 446 –614.

Milor, L. (1998). A tutorial introduction to research on analog and mixed-signal circuit testing. *IEEE TCAS-II*, **45**(10), 1389–1407.

Milor, L. and Sangiovanni-Vincentelli, A. (1994). Minimizing production test time to detect faults in analog circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **13**(6), 796 –813.

Milor, L. and Visvanathan, V. (1989). Detection of catastrophic faults in analog integrated circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **8**(2), 114 –130.

Nagi, N. and Abraham, J. (1992). Hierarchical fault modeling for analog and mixed-signal circuits. In *IEEE VLSI Test Symposium*, pages 96 –101.

Natarajan, V., Choi, H., Lee, D., Senguttuvan, R., and Chatterjee, A. (2008). EVM testing of wireless OFDM transceivers using intelligent back-end digital signal processing algorithms. *Test Conference, 2008. ITC 2008. IEEE International*, pages 1–10.

Papadimitriou, S., Kitagawa, H., Gibbons, P., and Faloutsos, C. (2003). LOCI: fast outlier detection using the local correlation integral. In *Data Engineering, 2003. Proceedings. 19th International Conference on*, pages 315 – 326.

Papoulis, A. and Pillai, S. (2001). *Probability, Random Variables and Stochastic Processes*. McGraw-Hill.

Pena, D. and Prieto, F. J. (2001). Multivariate outlier detection and robust covariance matrix estimation. *Technometrics*, **43**(3), pp. 286–300.

Perels, D., Haene, S., Luethi, P., Burg, A., Felber, N., Fichtner, W., and Bolcskei, H. (2005). ASIC implementation of a MIMO-OFDM transceiver for 192 Mbps WLANs. *Solid-State Circuits Conference, 2005. ESSCIRC 2005. Proceedings of the 31st European*, pages 215–218.

Petlenkov, E., Jutman, A., Nomm, S., and Ubar, R. (2008). Towards artificial intelligence based automatic adaptive response analyzer for high frequency analog bist. In *IEEE International Computational Intelligence for Measurement Systems and Applications*, pages 99 –104.

Saponara, S., L'Insalata, N. E., and Fanucci, L. (2008). Low-complexity FFT/IFFT IP hardware macrocells for OFDM and MIMO-OFDM CMOS transceivers. *Microprocessors and Microsystems, Elsevier*.

Scott, D. W. (2008). *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons.

Sebeke, C., Teixeira, J., and Ohletz, M. (1995). Automatic fault extraction and simulation of layout realistic faults for integrated analogue circuits. In *IEEE European Design and Test Conference*, pages 464 –468.

Senguttuvan, R., Bhattacharya, S., and Chatterjee, A. (2008). Fast accurate tests for multi-carrier transceiver specifications: EVM and noise. *VLSI Test Symposium, 2008. VTS 2008. 26th IEEE*, pages 175–180.

Singhee, A. and Rutenbar, R. (2010). Why quasi-monte carlo is better than monte carlo or latin hypercube sampling for statistical circuit analysis. *IEEE TCAD*, **29**(11), 1763 –1776.

Singhee, A. and Rutenbar, R. A. (2008). Statistical blockade: A novel method for very fast monte carlo simulation of rare circuit events, and its application. In *IEEE DATE*, pages 235–251.

Soma, M. (1991). An experimental approach to analog fault models. In *IEEE Custom Integrated Circuits Conference*, pages 13.6/1 –13.6/4.

Srinivasan, G., Chao, H.-C., and Taenzler, F. (2008). Octal-site EVM tests for WLAN transceivers on "very" low-cost ATE platforms. *Test Conference, 2008. ITC 2008. IEEE International*, pages 1–9.

Srinivasan, R. (2002). *Importance Sampling*. Springer.

Stratigopoulos, H.-G. and Makris, Y. (2005). Nonlinear decision boundaries for testing analog circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,*, **24**(11), 1760 – 1773.

Stratigopoulos, H.-G. and Mir, S. (2010). Analog test metrics estimates with PPM accuracy. In *IEEE ICCAD*, pages 241 –247.

Stratigopoulos, H.-G., Mir, S., Acar, E., and Ozev, S. (2009a). Defect filter for alternate rf test. pages 101–106.

Stratigopoulos, H.-G., Mir, S., and Bounceur, A. (2009b). Evaluation of analog/RF test measurements at the design stage. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pages 582 –590.

Stratigopoulos, H.-G. D., Drineas, P., Slamani, M., and Makris, Y. (2007). Non-RF to RF test correlation using learning machines: A case study. In *IEEE VLSI Test Symposium*, pages 9–14.

Taguchi, G. (1978). Performance analysis design. *International Journal of Production Research*, pages 521–530.

Turakhia, R., Benware, B., Madge, R., Shannon, T., and Daasch, R. (2005). Defect screening using independent component analysis on IDDQ. *IEEE VLSI Test Symposium*, pages 427–432.

Voorakaranam, R., Chakrabarti, S., Hou, J., Gomes, A., Cherubal, S., Chatterjee, A., and Kao, W. (1997). Hierarchical specification-driven analog fault modeling for efficient fault simulation and diagnosis. In *IEEE InternationalTest Conference*, pages 903 –912.

Yilmaz, E. and Ozev, S. (2008). Dynamic test scheduling for analog circuits for improved test quality. In *IEEE International Conference on Computer Design*, pages 227–233.

Yilmaz, E. and Ozev, S. (2009a). Adaptive test elimination for analog/rf circuits. In *IEEE Design Automation Conference*, pages 720–725.

Yilmaz, E. and Ozev, S. (2009b). Defect-based test optimization for analog/rf circuits for near-zero DPPM applications. *IEEE International Conference on Computer Design*, pages 313 –318.

Yilmaz, E. and Ozev, S. (2010). Adaptive test flow for mixed-signal/RF circuits using learned information from device under test. *IEEE International Test Conference*, pages 1–10.

Yilmaz, E., Ozev, S., and Butler, K. (2011). Adaptive multidimensional outlier analysis for analog and mixed signal circuits. In *IEEE International Test Conference (ITC)*, pages 1 –8.