Dynamic Test Scheduling for Analog Circuits for Improved Test Quality

Ender Yilmaz and Sule Ozev
Department of Electrical Engineering
Arizona State University
Ender.Yilmaz@asu.edu, Sule.Ozev@asu.edu

Abstract—In this paper, we present an innovative test scheduling method to improve test quality and/or reduce test time for analog circuits. Our dynamic test scheduling approach predicts the fail probability of unmeasured specifications with the aim of passing statistically well-behaved chips early on so as to devote more resources to marginal devices. Results show that for a gain controlled LNA circuit, with 48 specification parameters, it is possible to achieve 67% improvement in test quality for the same test time or 19.2% test time reduction with the same test quality compared to the widely used set cover method.

I. INTRODUCTION

Demand for smaller and more functional devices leads to highly packed and hard to test chips. Test cost is one of the most important incremental cost which scales inversely with feature size and becomes a bottleneck in the production line. Analog circuits are primarily tested by measuring their specified parameters. For a typical analog/RF circuit, the number of such parameters can easily exceed hundreds [1]. Measuring all these specifications may require seconds on the tester, resulting in unacceptable test cost. It is crucial to reduce analog test time and test cost in order to reduce time-to-market. To keep a competitive edge, the common practice in the industry is to limit the test time by selecting a subset of specifications to measure. Since test time reduction has been the overriding concern in the industry, most prior work in analog test has concentrated on this issue.

Various methods focus on reducing the overall test time by shrinking the number of tests. Set-cover based techniques [1], [2] aim at selecting a subset of tests that provide adequate coverage for the circuit. The subset can be selected based on simulation or characterization data. Due to its simplicity and ease of use, this technique is the most commonly used in the industry. Indirect testing has also been proposed as a low-cost alternative to full specification based testing [3], [4], [5], [6]. In one form of indirect test, carefully crafted test inputs are used to relate the response of the circuit under test (CUT) to its specification using regression analysis. Another popular indirect test strategy is to perform outlier analysis based on easy-to-measure signals, such as the bias current [7] or voltage.

Another common thread in analog test research is to schedule the tests so that tests that are most likely to fail are conducted first [8], [9], [10]. This strategy can reduce overall test time for a stop-on-fail scheme, particularly when the yield is low.

While reducing the test time has been the overriding issue in analog test research, test quality concerns have recently been raised. In the digital test domain, widely accepted metrics, such as defective parts per million (DPPM) have been provided to the customers. Customers have started to demand similar quality metrics for analog circuits as well. As a result, it is now imperative to focus both on test time and test quality for analog circuits. In [11], definitions of test quality metrics, such as fault coverage, yield coverage and DPPM have been presented. However, there has been little to no work in reducing the DPPM levels for analog circuits without blowing up the test time.

Previous work on test selection treat each circuit in an identical fashion devoting equal amount of effort to circuits until they fail or pass the tests. However, circuits are drawn from a statistically diverse set. In our work, we take advantage of this diversity and devote less time to well-behaved circuits, i.e. circuits that reside around the mid-point of the statistical distribution, while devoting more time to marginal circuits. Marginal circuits cannot be ignored since they constitute a considerable portion of good circuits. Identification of good circuits residing in the marginal region is hard since they are just on the edge. Cost reduction algorithms may fail to achieve required resolution in order to distinguish good circuits among marginal circuits. In this work, we propose to adaptively devote more time and effort on marginal circuits in order to achieve a higher selection resolution and to pass good circuits without spending as much time. Thus, quality metric is improved considerably compared to methods that treat each manufactured circuit identically. Circuits adjacent to nominal values are safer to pass, therefore, less testing is required to gain confidence. However, circuits with specification parameters close to specification limits are crucial and require special consideration. Treating marginal circuits more cautiously is our principal notion for quality improvement.

We present a dynamic method for test selection. We schedule tests according to statistical information obtained off-line, from a circuit ensemble, and on-the-fly, from each individual CUT. Circuits are treated according to their state of severity in terms of identifiability. Marginal circuits are identified using on-the-fly statistics and tested until a certain level of confidence is achieved. Statistically good circuits , however, are passed with less care since they are less likely to fail. On-the-fly statistics are updated using on-line test results predicting the states of unmeasured specification parameters.
II. METHODOLOGY

Similar to prior work in analog test, the goal of our method is to select a subset of tests from the response of which we can infer the goodness of the circuit. However, unlike prior work, each circuit instance is treated in a distinct way based on its characteristics. We observe that for circuits that fall at around the nominal for their physical parameters, their specification parameters also typically fall around the nominal. We exploit this correlation to make an early pass decision on well-behaved circuits. For other circuits, we need more information to make a pass/fail decision and these circuits take more time. By unevenly dividing the test resources and spending more time on marginal devices, our goal is to increase product quality without increasing test time.

In order to achieve this goal, we first need to obtain the correlation information. This step in our technique is called training. Training can be achieved by simulation or characterization and is off the critical path of testing. During the training process, we also select a small number of seed tests. These are the parameters which do not display a high correlation around their nominal point. During the test process, we first apply these seed tests to collect initial information on the CUT.

Fig. 1 shows the flow of the proposed dynamic method. Each applied test gathers more information from the circuit. Using the results of a measurement, we update the information on the likelihood of a parameter to be within the specification limits. This update is based on the correlation of two specific points in the likelihood profile of each parameter. Once the information on all unmeasured specifications is updated, we perform test filtering. Test filtering aims at dropping the test of a specification parameter once we reach a certain threshold for confidence level that it will be satisfied. For the remaining tests, we do not have enough information to make a decision. To select the next test, we find the specification parameter which has the least chance of passing since this means we have the least amount of information on it. This process is continued until we can make a decision on the circuit. The proposed method is dynamic in the sense of test selection; test sequence is selected according to the statical characteristics of the CUT. Since circuit instances are different from one other, their test sequence may be different.

Our test method is composed of consecutive steps: training, which is conducted off the critical path, and dynamic test scheduling, which is on the critical path. For the training step, a training set is comprehensively tested to obtain information about the specification correlations. The size of training set can vary from a couple hundred of instances to a few thousand. The quality of our test method, as well as any test compaction method, is highly dependent on this section. Increasing the training set size improves the performance of the testing step. Testing algorithms presented in literature utilize information on training set to obtain covering test set, regression equations or ordering information. Our method, uses the correlations of the specification domain parameters as an input for the proposed testing method. Generated correlation information is then used to relate specific points of measured parameters to specific points of unmeasured parameters.

A. Training

In order to enable test scheduling, we first need to extract information on parametric relations using a small set of circuits, the training set. Note that, any test technique, except for exhaustive testing, requires some form of training to enable test compaction. In this sense, the training phase is present in all test techniques, traditional or otherwise. We use this training phase to obtain a different kind of information: correlations between specific points of each pair of specifications.

Likelihood profiles of specification parameters are inherently continuous. However, obtaining a perfect profile of specifications and updating them with incremental information is not feasible. Probability mass function model is employed to reduce the complexity by binning specification parameter value range. This binning process is illustrated in Fig. 4. The bins are obtained by diving the specification space into small ranges. The binning process is explained in more detail in the following section. The most important features of the method, representation and updating the likelihood of parameter values, are thus simplified. With this binning strategy, we are interested in the correlations between the bins of two distinct specifications. If one of the specifications is measured, the probabilities that the other specification fall into each of its bins can be updated. In order to store the correlation information, an (NM) x (NM) matrix is needed where M is the number of specifications and N is the number of bins as illustrated in Fig. 2.

To extract this information, for each bin \( b_k \) in specification \( S_i \) and for each bin \( b_p \) in specification \( S_j \), we generate two vectors \( A_{ik} \) and \( A_{jp} \) with the size \( n_{trial} \times 1 \) where \( n_{trial} \) is the size of the training set. For each circuit instance in the training set, the vector entries are assigned as "1" if the specification falls into the corresponding bin, and as "0" otherwise. The correlation of two bins of \( b_k \) and \( b_p \) of the specification \( S_i \) and \( S_j \) is calculated by evaluating the correlation coefficient between vectors \( A_{ik} \) and \( A_{jp} \).
procedure UPDATE POST-
Prob(corrM, priorProb, selSpec, selBin)

corrVect = corrMatrix ∀ spec, ∀ bin selSpec, selBin

for all corrVecti do
  if corrVecti > 0 then
    postProbselSpec, j = 1 − (1 − priorProbselSpec, j)(1 − corrVecti)
  else
    postProbselSpec, j = (1 + corrVectj) priorProbselSpec, j
  end if
end for

normalize(postProb)
return postProb
end procedure

B. Updating the Likelihood of Bins

Once a measurement is taken, we can update the probability information for all unmeasured specifications. This process can be best explained through an example. Suppose that we measure a specification ($S_i$) and it falls into a particular bin ($b_k$). We now wish to update the probability that another specification ($S_j$) will fall into one particular bin ($b_p$). There are three cases to consider:

1. $b_k$ and $b_p$ are positively correlated: based on the observation that $S_i$ falls into $b_k$, there is now a greater chance that $S_j$ will fall into $b_p$.
2. $b_k$ and $b_p$ are negatively correlated: based on the observation that $S_i$ falls into $b_k$, there is now less of a chance that $S_j$ will fall into $b_p$.
3. $b_k$ and $b_p$ are not correlated: the chances of $S_j$ falling into $b_p$ remain unchanged.

In order to incorporate these observations in a systematic way, we use the following update formula:

\[
postProb_j = \begin{cases} 
1 - (1 - priorProbselSpec, j)(1 - corr_j) & , corr_j \geq 0 \\
(1 + corr_j)(priorProbselSpec, j) & , corr_j < 0 
\end{cases}
\]

The update procedure is given in Fig. 3. Initially, each bin for the specification parameters are equally likely. The likelihood profile reshapes after each measurement. The control variables of our proposed method are the number of bins and the confidence value to make a decision. As we will show in the results section, these parameters adjust the trade-off between test time and DPPM.

C. Test filtering

Once we have certain confidence that an unmeasured parameter will fall between its specification bounds, we can simply assume that the CUT has passed that specification and remove that specification from the test list. In order to do so, we use the updated likelihood profiles for each unmeasured specification.

In order to find the confidence level, we sum up the probabilities of the bins between the specification bounds. This simply gives us the probability that the CUT will be within that region even if we cannot know the exact value of that specification parameter. As an example, Fig.
Fig. 5 shows profiles of two specifications. Probabilities that are summed in this figure are highlighted bars, others are assumed to be out of specifications. Note that there exists a margin between specification boundaries and the outermost highlighted bars. This margin is included intentionally to reduce the misclassification. If there is a greater chance that the parameter falls into this region, further testing is needed. Note that with our technique, the probability of misclassifying good circuits is nil since we never reject the devices based on the likelihood parameter.

D. Selection of the Confidence Level Threshold

Confidence level threshold is one of the variables in our technique that controls the trade-off between test time and DPPM levels. Increasing the confidence threshold will make it harder to pass tests and will reduce the probability of misclassifying a bad circuit as good. As a result, the DPPM level will decrease. However, test time will increase since more information would need to be extracted from the circuit. Reducing the confidence level threshold will make the tests pass easier, reducing the number of applied tests. On the down side, test quality suffers since the probability of misclassification increases. In the results section, we will show how this confidence level threshold relates to the other parameters.

E. Test Selection

Since the principal idea of the proposed method is to obtain maximum information with least number of tests, selection of next test is performed by choosing the test of a specification that contains least amount of information in its likelihood profile. Consequently, information about CUT is adaptively determined through selecting most information bearing likelihood profile at each iteration. In order to determine the specification, on which we have the least amount of information, we use a heuristic metric. This metric is the standard deviation of the current likelihood profile of that unmeasured parameter. The intuition for selecting this metric is as follows. If we know a parameter’s value exactly, the likelihood profile will have a “1” corresponding the bin where the parameter falls and all other likelihoods are 0. As a result, the standard deviation of the likelihood profile will be zero. At the other extreme, if we have not obtained and updated any information on that parameter, its likelihood profile has not changed. Thus, at this point, the standard deviation of its likelihood profile will be highest.

For instance, profile shown in Fig. 5(b) is less informative compared to Fig. 5(a). Among these two tests, 5(b) qualifies to be the next test since less information is built in its profile, which implies that it is less correlated to the already measured specifications. Selecting less informative profile in each iteration allows us to determine and test relatively independent specifications and drop correlated ones if they build required confidence.

Iteration of test, update, select-next-test continues until all candidate tests are applied or dropped. Fig. 6 illustrates the flow of our methodology. The process starts with the first column on the left which contains 10 specifications. Crossed specifications show the ones that are selected to be tested and gray ones shows specifications which are dropped out of consideration thanks to adequate confidence level that they are satisfied. In the first iteration, measuring 3rd specification results in passing of first two tests. The rest of the specifications are propagated to the next iteration. In the second iteration, specification number 6 is selected to be tested and specifications 8-9 are passed as a result. The algorithm runs for two more iterations until all the tests are exhausted in the test set.

III. OPTIMIZED BINNING

In this section, we describe the implemented binning scheme and how it is obtained. Binning can be defined as partitioning a continuous range into a finite number of regions (bins). Our goal is to find the number of bins and determine boundaries of these bins on the parameter space. For \( N \) bins, \( N - 1 \) boundary points should be determined to achieve the optimal bin configuration. Obtaining \( N - 1 \) real valued boundary points is not practical due to evaluation cost of binning configurations. Therefore, an intuitive binning methodology is employed to obtain optimized boundary configurations with less computational effort.

The proposed binning method is to partition the area under a function into shapes with equal area. For instance as shown in Fig. 4(b), a flat function results in uniform binning of the parameter range. Problem of finding appropriate boundary points is transformed into finding a function that will result in optimal binning. Although latter one seems to be harder,
its complexity can be reduced by using some simplifications. The binning function is assumed to be piecewise linear and it contains finite number of pieces, so it can be represented using a finite number of points. The function is then built by simply connecting these points. The proposed method can be used to intuitively adjust bin boundaries. The specification under consideration is binned more intensely in regions where the binning function has a high value. Thus, we can adjust the regions that should be finely or coarsely binned. Positions of defining points of the binning function are selected to be some predefined critical points: minimum, maximum, mean, spec. min, spec. max, standard deviation or combination of these parameters. Fig. 4(c) shows a five-point function, where critical points are selected to be minimum, spec. min, mean, spec. max, and maximum. Fig. 4(d) shows a binning function which splits the regions around specification boundaries, while Fig. 4(e) shows a function that bins only the region between spec. min and spec. max, where middle region is binned more intensely.

Binning functions given in Fig. 4(c) and (d) are enumerated for their different function values at their defining points. In order to make the problem affordable, the function range is confined to have values [0,1] and discretized in finite number of values. For instance, if a function has 5 defining points and 3 possible values (0, 0.5, 1.0) for binning function, then 3<sup>5</sup> configurations of that function are enumerated to find the best binning function among candidates. Fig. 7 illustrates enumeration points of a 5-point binning function. Evaluation results show that the best configuration for binning is given in Fig. 4(e). Moreover, evaluation results degrade if middle region is coarsely binned. Outcomes of evaluations imply that middle region contains more valuable data and should be binned intensely in order to exploit this feature. Outer regions on the other hand can be binned loosely. This binning function is therefore selected to be used throughout the results section.

### IV. Results

The Proposed testing method is implemented for a variable gain LNA with 48 specifications. These specifications are gain, bandwidth, center frequency, 3db frequencies, offset, input impedance, output impedance, IIP3, 1db compression, power consumption and noise, defined for 4 different gain levels. Passive components are injected 3% variation while 1% variation is injected for parameters of active components (Width, Length).

Performance of the proposed method is compared to the set cover method, which is one of the most commonly used test methods in industry. ILP formulation has been proposed as a possible solution to find the optimal set cover [12]. The set-cover algorithm implemented in this work is chosen to be heuristic due to computational considerations of ILP. In this algorithm, incrementally the best covering test is selected in each iteration until all failing circuits can be classified. In the comparisons, test time is simply evaluated by the number of tests.

In order to enable the evaluation process, we generate 60k circuit instances by Monte-Carlo sampling and simulate them. We randomly select a subset of the circuit instances for training (both for our technique and for the set cover technique) and use the remaining instances to evaluate test time and DPPM. In order to ensure that the results do no coincide with an exceptionally good or bad point, we repeat this process 10 times by selecting a disjoint set of training instances.

#### A. Primary Results and Consistency

For the training set size of 5k circuits and 10 bins, Table I shows the DPPM comparison of the proposed method and set-cover method. The confidence levels of our technique are adjusted so as to equate the test time for a fair DPPM comparison. DPPM is calculated by extrapolating from a set of 25k samples, a disjoint set of 30k samples, and from the combined set of 55k samples to ensure consistency. The last row of Table I shows the standard deviation of the obtained DPPM levels. Table I clearly shows that for equal test time, our technique considerably improves test quality over the set-cover technique.

We also equate the DPPM levels of our technique to that of the set-cover technique to see the effect on test time. Table II shows this comparison. We observe that our technique also improves test time consistently for equal test quality. The low standard deviation also means that our technique is much less dependent on the characteristics of the training population.
TABLE I
DPPM COMPARISON FOR THE SAME TEST TIME

<table>
<thead>
<tr>
<th></th>
<th>20k</th>
<th>35k</th>
<th>55k</th>
</tr>
</thead>
<tbody>
<tr>
<td>method</td>
<td>ours</td>
<td>SC</td>
<td>ours</td>
</tr>
<tr>
<td>mean</td>
<td>308</td>
<td>946</td>
<td>310.56</td>
</tr>
<tr>
<td>std</td>
<td>350.81</td>
<td>418.49</td>
<td>330.62</td>
</tr>
</tbody>
</table>

TABLE II
TIME COMPARISON FOR THE SAME DPPM LEVEL

<table>
<thead>
<tr>
<th></th>
<th>20k</th>
<th>35k</th>
</tr>
</thead>
<tbody>
<tr>
<td>method</td>
<td>ours</td>
<td>SC</td>
</tr>
<tr>
<td>mean</td>
<td>20.52</td>
<td>26.06</td>
</tr>
<tr>
<td>std</td>
<td>0.29</td>
<td>1.66</td>
</tr>
</tbody>
</table>

B. Effect of Training Set Size

We also evaluate the effect of training set size on test metrics as shown in Fig. 9. Three different training sets are used to obtain likelihood profiles with sizes 1k, 5k, and 10k instances. The figure shows DPPM and time results for 10 bin configuration and various confidence levels ranging from 80% to 100%.

Not surprisingly, increasing the training set size boosts test quality. For instance DPPM of 94% confidence level for 10k training set is almost equal to the set cover value; they are used as a reference point for comparison. Reducing training set size to 1k degrades DPPM by 260%. DPPM of set cover degrades even more, by 6680%, which suggests that set cover method is more sensitive to training set size. Critical points of Fig. 9 are give in Table III and Table IV to compare DPPM improvement and time reduction respectively. In Table III DPPM levels of two methods are compared for the same test time, while in the Table IV test times are compared for the same level of DPPM.

We observe from Fig. 9 that DPPM is highly dependent on the confidence level. For a confidence level of 100%, DPPM drops to zero since this means that all specifications need to be measured. Conversely, the test time is not much dependent on the training set size. It should be noted that for techniques, such as set cover test time increases as the training set size increases. Also, observe from Fig. 9 that, the while the DPPM is almost exponentially dependent on confidence threshold level, the test time is almost linear to it. These features enable setting the confidence level for a desired level of DPPM or test time and evaluating the effect on the other parameter.

TABLE III
DPPM COMPARISON BETWEEN OUR METHOD AND THE SET COVER METHOD FOR EQUAL TEST TIME FOR VARIOUS TRAINING SET SIZES

<table>
<thead>
<tr>
<th></th>
<th>1k</th>
<th>5k</th>
<th>10k</th>
</tr>
</thead>
<tbody>
<tr>
<td>setCover</td>
<td>5423.64</td>
<td>1012.22</td>
<td>79.99</td>
</tr>
<tr>
<td>ours</td>
<td>3500</td>
<td>470</td>
<td>0</td>
</tr>
</tbody>
</table>

C. Confidence Level Threshold and Binning

We also evaluate the combined effect of confidence level and the number of bins. Fig. 10(a) shows the combined effect of the number of bins and confidence level threshold on DPPM and the Fig. 10(b) shows the same effect on test time. One trend seen in the figure is the increase of the DPPM level as the number of bins are increased for the same training set. Test time, however, decreases as the number of bins are increased. The increase in the DPPM level and the reduction in test time can be attributed to the noise in the statistical information. Noise in the correlation information increases as the number of instance falling in each bin reduces. These variables provide us a way of controlling the important test metrics: DPPM and test time to the desired levels. Based on the desired DPPM goals, the confidence level and number of bins can be adjusted to achieve the best test time.

D. Delay of Identification

As we have mentioned, in order to decide whether the circuit under test is faulty or not, decision is delayed until
In this work, we present a dynamic test selection method for improving test quality. Primary innovation introduced in this paper is the adaptive test selection based on on-the-fly data for each individual CUT to devote more effort on marginal circuits and pass good circuits without vesting time. The proposed method guides the test selection mechanism using on-the-fly obtained likelihood profiles and correlation information data gathered prior to testing to determine an optimized test sequence for each CUT. Our goal is to obtain the confidence to pass the circuit as fast as possible based on likelihood profiles or fail it based on direct measurement.

Experimental results show that our method can improve test quality, as measured by defective parts per million, by 67% when compared to the widely used set-cover based test selection method with the same test time. In a similar manner, our method can improve test time by 20% compared to the set cover method when DPPM levels are equal. Another important feature of this method is that it offers a trade-off between time and quality. Thus, it is possible to choose different levels of improvement giving chance to control both of the test metrics, time and quality.

**REFERENCES**


