Transactions Briefs

RF Specification Test Compaction Using Learning Machines

Haralampos-G. Stratigopoulos, Petros Drineas, Mustapha Slamani, and Yiorgos Makris

***Abstract*—We present a machine learning approach to the problem of RF specification test compaction. The proposed compaction flow relies on a multi-objective genetic algorithm, which searches in the power-set of specification tests to select appropriate subsets, and a classifier, which makes pass/fail decisions based solely on these subsets. The method is demonstrated on production test data from an RF device fabricated by IBM. The results indicate that machine learning can identify intricate correlations between specification tests, which allows us to infer the outcome of all tests from a subset of tests. Thereby, the number of tests that need to be explicitly carried out and the corresponding cost are reduced significantly without adversely impacting test accuracy.**

***Index Terms*—Artificial intelligence, circuit testing, RFICs.**

I. Introduction

Specification testing still remains the only acceptable industrial practice for RF devices. In this approach, the performances of the device are measured one by one and are verified against the specification limits. Yet the high cost of RF automatic test equipment (ATE) and the lengthy test times involved have resulted in intensified efforts and interest in reducing the number and types of specification tests that are performed during production testing. A plausible direction towards decreasing cost, akin to test compaction practices in digital circuits, is to identify and eliminate information redundancy in the set of tests, thereby relying only on a subset of them in order to reach a pass/fail decision [1]–[6]. Such redundancy is likely to exist since groups of performances refer to the same portion of the device and are subject to similar process imperfections. However, it is highly unlikely that it will manifest itself in a coarse and easily observable form of superfluous tests that can be summarily discarded. Hence, more advanced statistical analysis methods are likely to be required.

Our approach to the problem of RF specification test compaction entails two components, namely a feature selection algorithm for searching in the power-set of specification tests and a prediction model for making pass/fail decisions based solely on a select subset. In particular, we use a multi-objective genetic algorithm (GA) for feature selection and we build the prediction model based on a binary classifier. The search progresses towards a low-cost, low-dimensional subset based on which the classifier predicts correctly the pass/fail outcome of the complete specification test suite. The proposed compaction methodology is evaluated on a zero-IF down-converter for cell-phone applications designed in RFCMOS technology and fabricated by IBM. The main conjecture drawn from this case study is that a relatively small number of only non-RF specification tests (i.e., digital, dc, and low frequency) are shown to suffice for predicting correctly the pass/fail label of a very large percentage of devices, a scenario that would eliminate the need for RF ATE. Moreover, the addition of a few RF specification tests ameliorates this small prediction inaccuracy and results in very powerful prediction models, a scenario that would still require RF ATE but would reduce the time that a device spends on it.

The remainder of this paper is organized as follows. In Section II, we describe the basic principle of the proposed method, we introduce the classifiers, we discuss feature selection using GAs, and we present the model that is used to evaluate the cost of a specification test subset. In Section III, we present our case study. Experimental results are provided in Section IV and the conclusion is drawn in Section V.

II. Machine-Learning-Based Test Compaction

A. Basic Principle

Let $S = [s_1, \ldots, s_d]$ denote the set of specification tests for the device under test. A set of N devices is subjected to S. For each device, we record s_k, $k = 1, \ldots, d$, and the resulting ground truth pass/fail label. We split the set of N devices in a training and validation set. We assume that the pass/fail labels are known only for the devices in the training set; target labels for the devices in the validation set are assumed to be unknown and they are only used to estimate the test error.

A GA is used to search in the power-set of the 2^d subsets of S. Each visited subset $S' \subseteq S$ is assigned a fitness value based on two criteria: (a) the associated cost $C(S')$ and (b) the incurred test error $\epsilon_r(S')$ when new devices coming out of production are subjected only to S'. Pass/fail assignments based solely on S' are done as follows. Devices that fail one or more tests in S' are discarded outright. Devices that pass all tests in S' are presented to a classifier that establishes a binary mapping of the form $f : S' \rightarrow (\text{pass,fail})$. This mapping is learned in a training phase which employs the devices in the training set. The error $\epsilon_r(S')$ is defined as the percentage of devices in the validation set that pass all tests in S', but are misclassified by the mapping f.

The GA explores the tradeoff $\epsilon_r - C$ with the aim to converge to the Pareto frontier, i.e., the set of subsets of S that are not strictly dominated[1] by another subset of S. The search evolves until an objective for the fitness criteria is met or until a large number of iterations is completed, implying that further optimization of the fitness criteria is not possible.

Manuscript received June 27, 2008; revised September 23, 2008. First published July 21, 2009; current version published May 26, 2010. This research was supported in part by the Commission of European Communities (MIRG-CT-2007-209653), the National Science Foundation (NSF ECS-0622081 and CCF-0702522), and the Semiconductor Research Corporation (SRC 1632).

H.-G. Stratigopoulos is with TIMA Laboratory, CNRS-Grenoble Institute of Technology-Université Joseph Fourier, Grenoble 38031, France (e-mail: haralampos.stratigopoulos@imag.fr).

P. Drineas is with the Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180-3522 USA (e-mail: drinep@cs.rpi.edu).

M. Slamani is with the IBM Wireless Test Development Group, Essex Junction, VT 05452 USA (e-mail: slamanim@us.ibm.com).

Y. Makris is with the Departments of Electrical Engineering and Computer Science, Yale University, New Haven, CT 06520-8285 USA (e-mail: yiorgos.makris@yale.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2009.2017196

[1]A subset S^i with fitness criteria (ϵ_r^i, C^i) strictly dominates a subset S^j with fitness criteria (ϵ_r^j, C^j) if (a) $\epsilon_r^i \leq \epsilon_r^j$ and $C^i < C^j$ or (b) $C^i \leq C^j$ and $\epsilon_r^i < \epsilon_r^j$.

B. Classifiers

1) k-Nearest-Neighbors: k-Nearest-Neighbors (k-NN) algorithms constitute perhaps the simplest nonlinear classifiers [7]. Given a pattern S'_u, whose target (pass or fail) label is unknown, we examine its k nearest neighbors in a training set (for an odd value of k), and assign S'_u to the label having the largest number of representatives amongst the nearest neighbors. This amounts to a majority vote in the neighborhood of S'_u. Picking the value of k is data-dependent. Typically, increasing k returns better results, until a point of diminishing returns is reached. Another relevant parameter is the nearness metric. In our experiments, we use Euclidean distance to determine pattern proximity.

2) Ontogenic Neural Network (ONN): The second classifier that we employ in this study is an ONN [8]. It is trained to allocate a hyper-surface separating the populations of functional and faulty devices in the subspace defined by S'. When a new pattern S'_u is presented to the trained neural network, it makes a pass or fail decision depending on the footprint of S'_u with respect to the hyper-surface.

The ONN can draw arbitrarily nonlinear hyper-surfaces (convergence to zero classification error on a training set is proven) and, in principle, it is a universal classifier without requiring *a priori* knowledge about the parametric form of the true separation hyper-surface. The learning algorithm searches for small network solutions first and, thus, offers a potential for discovering a near-minimal network that suitably solves the separation problem. In other words, the network architecture is not fixed, but is constructed successively during learning. Thus, the ONN has a clear advantage over feed-forward neural networks, which require a computationally expensive trial-and-error approach to select the appropriate network architecture, and over support vector machines, which require the *a priori* definition of a kernel. Given that each layer in the ONN comprises a single perceptron, training of a few layers can be done very fast. This is a particularly attractive attribute since an ONN is trained each time a new test subset S' is visited during the course of the GA so as to compute $\epsilon_r(S')$.

C. Genetic Algorithm

The use of GAs for selecting features from a high-dimensional set is originally proposed in [9]. GAs maintain a population of chromosomes of fixed size. In our case, chromosomes are bit strings of length equal to the number of specification tests, where the jth bit is set to 1 if the jth test is present in the subset and 0 otherwise. Starting with a base population, new chromosomes are generated using the mutation and crossover operators. In crossover, parts of two different parent chromosomes are mixed to produce an offspring. In mutation, bits of a single parent chromosome are randomly perturbed to create a child. At the end of each generation, each chromosome is evaluated to determine its fitness criteria. Only the fittest chromosomes are likely to survive and breed into the next generation. GAs evolve with the juxtaposition of bit templates, quickly optimizing the target fitness criteria. In this work, we use a multi-objective GA, called NSGA-II [10], to jointly optimize in one simulation run both the prediction error of the classifier and the test cost. For this purpose, NSGA-II has a diversity preserving mechanism that ensures a good spread of the Pareto frontier.

D. Test Cost Model

We consider the general case where a specification test set comprises tests that require different test instrumentation and execution times. We group the specification tests in S according to their type into M test groups. If we denote by n_i the number of tests in test group i, then $d = n_1 + \cdots + n_M$ is the total number of specification tests. Let $T(S)$ and $C(S)$ be the baseline test time and test cost per second, respectively, when all d specification tests are considered. Then

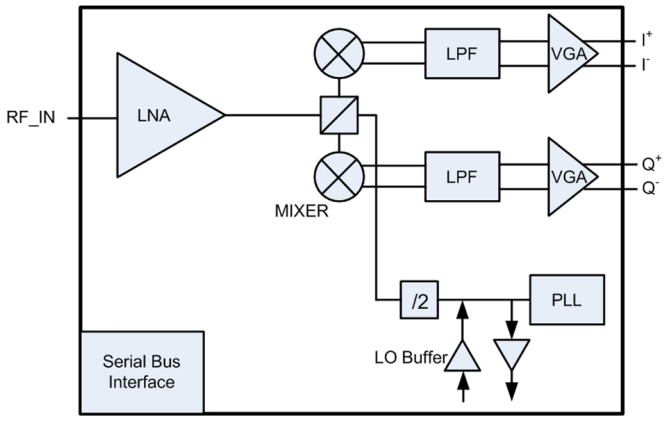

Fig. 1. Block diagram of the RF device.

$$C(S) = \sum_{i=1}^{M} (c_i C_{RF})\,(t_i T(S)) = C_{RF} T(S) \sum_{i=1}^{M} c_i t_i \qquad (1)$$

where t_i is the relative test time contribution of test group i with respect to $T(S)$ and c_i is the relative test cost per second of test group i with respect to the test cost per second of RF ATE, denoted here by C_{RF}. Let now $x_{ik} = 1$ if test k in the test group i is present, and $x_{ik} = 0$ otherwise. Let also t_{ik} denote the test time of test k in the test group i. The test cost of a subset S' is given by

$$C(S') = \sum_{i=1}^{M} \left(c_i (1 - \overline{x_{i1}} \cdot \ldots \cdot \overline{x_{in_i}}) C_{RF} \sum_{k=1}^{n_i} t_{ik} x_{ik} \right) \qquad (2)$$

where the symbol $\cdot$ denotes the logic AND. Assuming that $t_{ik} = t_i T(S)/n_i$, the preceding equation becomes

$$C(S') = C_{RF} T(S) \sum_{i=1}^{M} \left(\frac{c_i t_i}{n_i} \sum_{k=1}^{n_i} x_{ik} \right). \qquad (3)$$

The normalized test cost fitness criterion of a subset S' is given by

$$\tilde{C}(S') = \frac{C(S')}{C(S)} = \frac{\sum_{i=1}^{M} \left(\frac{c_i t_i}{n_i} \sum_{k=1}^{n_i} x_{ik} \right)}{\sum_{i=1}^{M} c_i t_i}. \qquad (4)$$

III. Case Study

Our case study is a zero-IF down-converter for cell-phone applications designed in RFCMOS technology and fabricated by IBM. The block diagram is shown in Fig. 1. This device is characterized by 143 specification tests, of which 72 require RF ATE and 71 do not. The different groups of specification tests, the number of tests in each group, as well as the relative test time and cost of each group are given in Table I. Our data set contains the measured specification tests for $N = 4450$ devices that were collected from four different lots within a period of six months. The corresponding specification limits are used to assign to each device a pass or fail status label. In total, 4142 devices pass all specification tests while 308 devices fail at least one test. The baseline

TABLE I
COST INFORMATION

Test Group	*Test Type*	n_i	t_i	c_i
SBI_tests	Digital	25	6.0%	40%
SBI_tests Supply_currents	DC	34	13.3%	
DacTests	Mixed Signal	6	3.3%	60%
Lock_vco		6	13.0%	
Lock_vco	RF	1	1.1%	100%
Filter_tests		20	13.3%	
Mixer_tests		43	30%	
LNA_tests		8	20%	

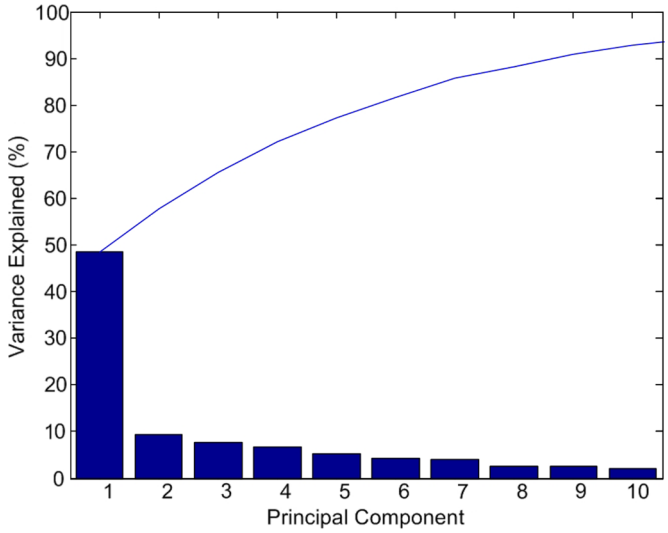

Fig. 2. Percentile variability explained by each principal component.

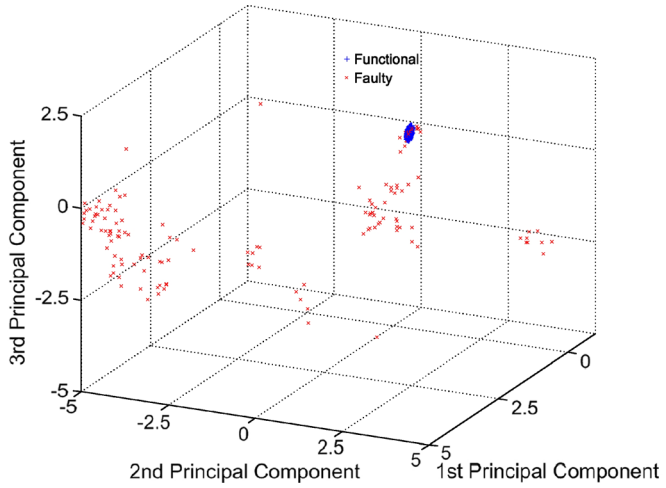

Fig. 3. Projection of devices onto the top three principal components.

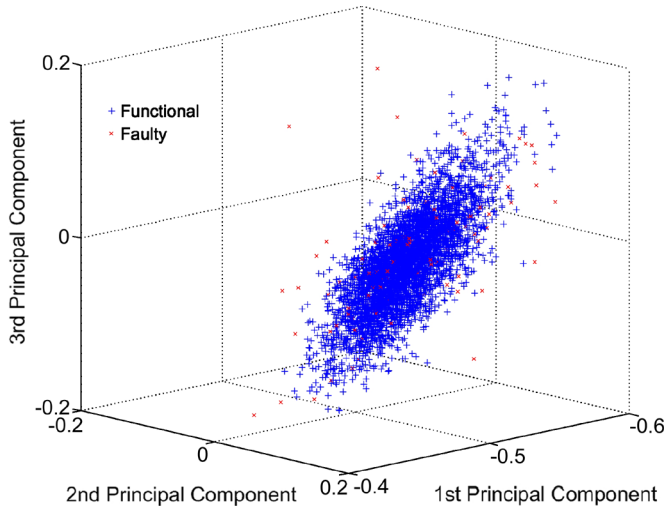

Fig. 4. Zoom in the core of functional devices in the 3-D plot of Fig. 3.

test time includes handling time, instrumentation settling time, electrical test time, data transfer time from tester instrumentation (such as digitizer) to tester computer, FFT computing time, and phase-locked loop (PLL) settling time.

To gain some intuitive understanding of our data set, we first perform a principal component analysis (PCA), which is a commonly used learning method for unsupervised dimensionality reduction [7]. PCA projects the original data onto a new set of orthonormal axes (e.g., principal components). The formed variables corresponding to the principal components have variances in decreasing order. Fig. 2 illustrates the percentile variability explained by each principal component. It can be observed that retaining the first 10 principal components suffices to capture over 95% of the information content in the data. This redundancy is key in accurately predicting pass/fail using only a subset of specification tests.

Fig. 3 plots the 4450 devices on the coordinate system of the top three principal components. Even in this rather primitive visualization, it is evident that the majority of the faulty devices are easily detected since their patterns are very distant from the dense core of functional devices. However, in Fig. 4, which zooms in on this core, it can be observed that there exist faulty devices whose patterns are interwoven with the functional ones. Separating such devices in just three dimensions seems difficult and one can only hope that this will be achieved by adding dimensions. Moreover, it can be observed that there are faulty devices whose patterns fall in close proximity to the core. These faulty devices are critical and should appear in the training set in large numbers, in order to learn a good representation of the true separation boundary.

IV. RESULTS

A. Experimental Setup

The training and validation sets comprise a fixed number of $(3/4)N$ and $(1/4)N$ devices, respectively. Notice that, on average, there will be 77 faulty devices in a validation set. The test subset selection terminates when the GA completes $n_g = 50$ generations. $n_c = 2d'$ chromosomes (i.e., specification test subsets) are being evaluated in each generation, where d' is the string length of the binary-coded chromosome. The crossover and mutation probabilities are set to $p_c = 0.9$ and $p_m = 1/d'$, respectively. To reduce the variance of the reported test error, the classifier is trained L times for each visited subset S' using different splits of the devices in training and validation sets (devices are sampled uniformly at random). We report the average test error

$$\hat{\epsilon}_r(S') = \frac{1}{L}\sum_{i=1}^{L} \epsilon_{r_i}(S') \tag{5}$$

where ϵ_{r_i} is the test error of split i.

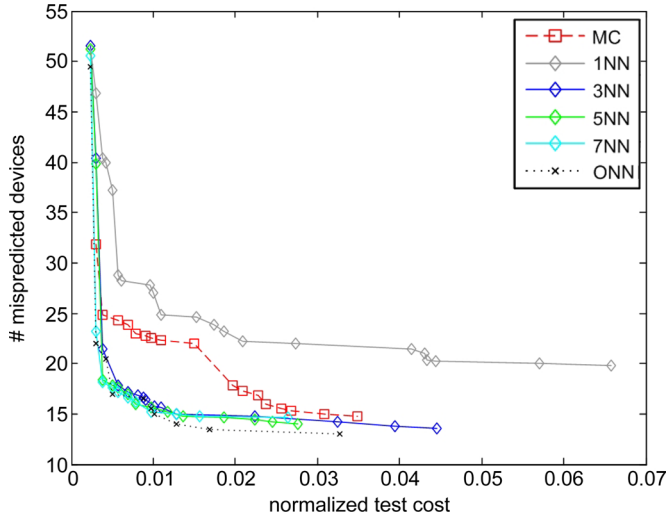

Fig. 5. Test error (i.e., average number of misclassified devices in the validation set) versus normalized test cost [defined in (4)] when using only non-RF specification tests.

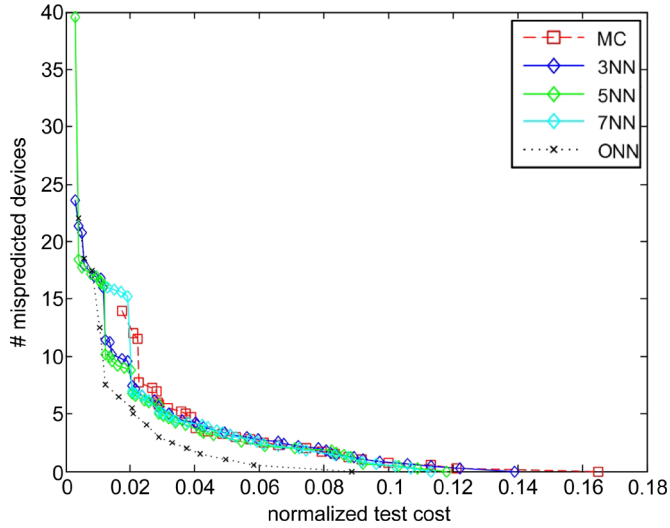

Fig. 6. Test error (i.e., average number of misclassified devices in the validation set) versus normalized test cost [defined in (4)] when adding RF specifciation tests to the best selected subsets (in terms of test error) of Fig. 5.

As a basis for comparison, we also investigate a simple weighted maximum-cover (MC) formulation of the compaction problem. In this approach, the prediction model is trivialized, the classifier is removed from the flow and the pass or fail decision is reached by simply verifying the kept specification tests in S', while ignoring the rest of specification tests. In this scenario, for a given cost, the GA will try to identify the subset that covers the maximum number of faulty devices. Other possible formulations of the maximum-cover problem are based on integer linear programming (ILP) [6] and Boolean minimization [11].

B. Selecting Only Non-RF Specification Tests

We first consider the set of specification tests whose execution does not require RF ATE ($d' = 71$). The proposed test compaction flow is evaluated using k-NN and ONN as the underlying classifiers. We also evaluate the simplistic MC formulation where there is no classifier involved. The results are plotted together in Fig. 5 which shows the Pareto frontiers that are found by the GA. Each point in the Pareto frontier corresponds to a test subset which can achieve this tradeoff. Notice that the misprediction rate for the MC corresponds to test escapes only while, in the case where classifiers are used, it could partially correspond to yield loss.

As can be observed from the MC curve, using only non-RF specification tests, we can detect 62 out of the 77 failing devices, i.e., the minimum number of mispredicted devices is 15. This is achieved at a small cost of 0.035. The k-NN formulation outperforms MC for $k > 1$. For $k > 1$ the tradeoff curve is practically the same, so we only report the results for $k = 3, 5,$ and 7. Finally, ONN outperforms both k-NN and MC. As may be observed from the ONN curve, the minimum number of mispredicted devices is 13.

A general remark that can be made is that the classifiers can correctly classify faulty devices based on a subset, despite the fact that none of the tests in the subset violates its specification. For example, without loss of generality, consider subsets with a fixed cost of 0.01. If a straightforward comparison to the specifications is made, the best subset would mispredict 23 out of the 77 faulty devices (MC curve). On the other hand, there exists a (possibly different) subset which, if processed by a classifier, will result on an average of 15 mispredictions (k-NN and ONN curves). Thus, considering the worst case scenario where all these mispredictions concern faulty devices, the classifiers can correctly classify an additional 8 faulty devices for a fixed cost. This means that the classifiers encode correlations between tests which, otherwise, would be very difficult to deduce by just observing the test data logs.

TABLE II
NUMBER OF TESTS FROM EACH TEST GROUP THAT PARTICIPATE IN THE BEST SUBSETS (IN TERMS OF MINIMUM TEST ERROR) FOR THE ONN

Test Group	*Test Type*	n_i *(section IV-B)*	n_i *(section IV-C)*
SBI_tests	Digital	1	0
SBI_tests Supply_currents	DC	4	2
DacTests	Mixed Signal	2	1
Lock_vco		1	0
Lock_vco	RF	0	0
Filter_tests		0	2
Mixer_tests		0	4
LNA_tests		0	1

C. Adding RF Specification Tests

Next, we examine the prediction improvement that can be obtained by adding RF specification tests to the best (in terms of minimum test error) identified subsets of non-RF specification tests. Formally, let S_i^{n-RF} be the selected subset for method i in Section IV-B, where $i = \mathrm{MC}$, k-NN, and ONN. Let also S^{RF} be the subset of RF specification tests. The starting sets in this step are $(S_{k-NN}^{n-RF} \bigcup S_{MC}^{n-RF}) \bigcup S^{RF}$, $k = 3, 5, 7$, $(S_{ONN}^{n-RF} \bigcup S_{MC}^{n-RF}) \bigcup S^{RF}$, and $S_{MC}^{n-RF} \bigcup S^{RF}$. We have taken the union of S_{k-NN}^{n-RF} and S_{ONN}^{n-RF} with S_{MC}^{n-RF} to ensure that we do not discard tests that uniquely detect failures. It is evident that some non-RF specification tests might be removed in this step on account of their correlation to a required RF specification test. 1-NN is not considered as it yielded significantly worse results than MC in the previous selection step.

The results are plotted in Fig. 6. As may be observed, there exists a subset of cost 0.165 that detects all faulty devices (MC curve). The k-NNs achieve the same objective (with no yield loss) for a subset of cost that ranges approximately between 0.115 and 0.14. However, in

the range of lower costs, the k-NN and MC curves practically coincide, meaning that k-NN does not offer a significant advantage over the simplistic solution. In contrast, ONN outperforms both MC and k-NNs. Zero test error is achieved for a subset of cost 0.09.

Table II shows the number of kept specification tests from each test group in the case of ONN. The third column refers to the best non-RF subset, which achieves an error of 13 mispredicted devices. The forth column shows the additional RF specification tests that suffice to reduce this error down to zero. Note that some specification tests in the best non-RF subset have been eliminated on account of their correlation to the selected RF specification tests. In particular, the digital tests as well as the tests to examine the lock of the voltage control oscillator (VCO) have been deduced to be redundant. Intuitively, the latter tests have been eliminated on account of their correlation to the tests of the mixer.

V. Conclusion

Analysis of historical production test data from an RF device reveals that specification tests comprise significant redundancy. This redundancy can be exploited to build prediction models for reaching pass or fail decisions based on a reduced-size set of specification tests. To this end, advanced machine learning methods, such as the combination of a multi-objective GA and an ONN, achieve excellent results and demonstrate great potential for reducing test cost through specification test compaction. For a specification test subset of cost $0.09 \times C(S)$, where $C(S)$ is the cost of the complete specification test suite, our results indicate that the ONN can predict correctly the pass or fail label of all devices in our data set. Moreover, this result is considered to be statistically significant since the data set was collected from four different lots within a period of six months.

References

[1] S. D. Huss and R. S. Gyurcsik, "Optimal ordering of analog integrated circuit tests to minimize test time," in *Proc. ACM/IEEE Des. Autom. Conf.*, 1991, pp. 494–499.

[2] L. Milor and A. L. Sangiovanni-Vincentelli, "Minimizing production test time to detect faults in analog circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits and Systems*, vol. 13, no. 6, pp. 796–813, Jun. 1994.

[3] J. B. Brockman and S. W. Director, "Predictive subset testing: Optimizing IC parametric performance testing for quality, cost, and yield," *IEEE Trans. Semicond. Manuf.*, vol. 2, no. 3, pp. 104–113, Mar. 1989.

[4] S. Biswas, P. Li, R. D. Blanton, and L. Pileggi, "Specification test compaction for analog circuits and MEMS," in *Proc. Des., Autom. Test Eur.*, 2005, pp. 164–169.

[5] S. Biswas and R. D. Blanton, "Statistical test compaction using binary decision trees," *IEEE Des. Test Comput.*, vol. 23, no. 6, pp. 452–462, Jun. 2006.

[6] H.-G. D. Stratigopoulos, P. Drineas, M. Slamani, and Y. Makris, "Non-RF to RF test correlation using learning machines: A case study," in *Proc. IEEE VLSI Test Symp.*, 2007, pp. 9–14.

[7] C. M. Bishop, *Neural Networks for Pattern Recognition*. London, U.K.: Oxford University Press, 1995.

[8] H.-G. Stratigopoulos and Y. Makris, "Error moderation in low-cost machine-learning-based Analog/RF testing," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 27, no. 2, pp. 339–351, Feb. 2008.

[9] W. Siedlecki and J. Sklansky, "A note on genetic algorithms for large-scale feature selection," *Pattern Recog. Lett.*, vol. 10, pp. 335–347, 1989.

[10] K. Deb, A. Pratap, A. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evolutionary Comput.*, vol. 6, no. 2, pp. 182–197, Feb. 2002.

[11] S. Biswas and R. D. Blanton, "Test compaction for mixed-signal circuits using pass-fail test data," in *Proc. IEEE VLSI Test Symp.*, 2008, pp. 299–308.