Investigations on the Use of Ensemble Methods for Specification-Oriented Indirect Test of RF Circuits

Hassan El Badawi, Florence Azaïs, Serge Bernard, Mariane Comte, Vincent Kerzérho, François Lefèvre

Title:
Investigations on the use of ensemble methods for specification-oriented indirect test of RF circuits

Authors:
H. El Badawi [1,2], F. Azais [1], S. Bernard [1], M. Comte [1], V. Kerzerho [1], F. Lefevre [2]

Affiliations:
[1] LIRMM, University of Montpellier, CNRS, 161 rue Ada, 34095 Montpellier Cedex, France
[2] NXP Semiconductors, 2 Esplanade Anton Phillips, 14000 Caen, France

Corresponding author:
Florence Azais
LIRMM, University of Montpellier, CNRS, 161 rue Ada, 34095 Montpellier Cedex, France
Tel: +33 467 418 625
Email: florence.azais@lirmm.fr

Abstract:
In order to reduce the costs of industrial testing of analog and Radio Frequency (RF) integrated circuits, a widely studied solution is indirect testing. Indeed, indirect testing is based on learning-machine algorithms to train a regression model that links the space of low-cost indirect measurements to the space of performance parameters guaranteed by datasheets, thus relaxing the constraints on expensive test equipment. This article explores the potential benefit of using ensemble learning in this context. Unlike traditional learning models that use a single model to estimate targeted parameters, ensemble-learning models involve training several individual regression models and combining their outputs to improve the predictive power of the ensemble model. Different ensemble methods based on bagging, boosting or stacking are investigated and compared to classical individual models. Experiments are performed on three RF performances of a LNA for which we have production test data and model quality is discussed in terms of goodness-of-fit, accuracy and reliability. The influence of the training set size is also explored. Finally, the efficiency of classical and ensemble models is compared in the context of a two-tier test flow that permits to tradeoff test cost and test quality.

Keywords: indirect testing, RF integrated circuits, machine-learning algorithms, ensemble methods, test efficiency

Investigations on the use of ensemble methods for specification-oriented indirect test of RF circuits

H. El Badawi [1,2], F. Azais [1], S. Bernard [1], M. Comte [1], V. Kerzerho [1], F. Lefevre [2]
[1] LIRMM, University of Montpellier, CNRS, 161 rue Ada, 34095 Montpellier Cedex, France
[2] NXP Semiconductors, 2 Esplanade Anton Phillips, 14000 Caen, France

***Abstract* — In order to reduce the costs of industrial testing of analog and Radio Frequency (RF) integrated circuits, a widely studied solution is indirect testing. Indeed, indirect testing is based on learning-machine algorithms to train a regression model that links the space of low-cost indirect measurements to the space of performance parameters guaranteed by datasheets, thus relaxing the constraints on expensive test equipment. This article explores the potential benefit of using ensemble learning in this context. Unlike traditional learning models that use a single model to estimate targeted parameters, ensemble-learning models involve training several individual regression models and combining their outputs to improve the predictive power of the ensemble model. Different ensemble methods based on bagging, boosting or stacking are investigated and compared to classical individual models. Experiments are performed on three RF performances of a LNA for which we have production test data and model quality is discussed in terms of goodness-of-fit, accuracy and reliability. The influence of the training set size is also explored. Finally, the efficiency of classical and ensemble models is compared in the context of a two-tier test flow that permits to tradeoff test cost and test quality.**

Keywords: indirect testing, RF integrated circuits, machine-learning algorithms, ensemble methods, test efficiency

1 Introduction

A circuit must be testable to be viable and for more than 50 years, the testing of integrated circuits has been incorporated into their development process. In the case of digital blocks, even though complexity has exploded over the decades, fault-oriented testing has allowed to limit the part of the testing costs of these blocks. On the other hand, for analog and RF blocks, even if they have evolved less in complexity than digital blocks, their testing costs have continued to increase. The main reason is that there is no recognized fault model for analog and RF blocks and therefore fault-oriented approaches are inadequate. In consequence, analog and RF circuits are tested with a specification-oriented approach, which relies on the measurement of the circuit performances and the verification of whether these performances comply with the datasheet. This approach ensures a satisfying test quality but the required measurements necessitate very expensive test equipment and long test time, which are responsible for the excessive testing costs.

Several approaches have been studied to avoid these direct performance measurements. All fault model-based solutions such as digital test techniques never achieved an acceptable level of test efficiency, even if recent solutions improve the effectiveness of these techniques [1]. Built-in-Self-Test or DFT (Design for Testability) solutions often reduce the required external resources [2], but have a significant silicon impact, and above all, do not allow performance to be measured with the same accuracy as external measurement instruments.

In this context, an interesting approach is to adopt an indirect test strategy based on machine-learning algorithms. The objective is to evaluate the circuit performances, not by the classical direct measurements but by measurements of other parameters, called Indirect Measurements (IMs), which only necessitate low-cost test resources and low test time. In a preliminary learning phase, both the conventional measurements of the device performances and the low-cost indirect measurements are performed on a set of training devices, in order to establish the correlation between the two types of parameters. Then, in the production testing phase, the performances of every new device are evaluated based solely on the low-cost indirect measurements, and the previously learned correlation.

This approach of indirect testing has been largely studied in the literature over the past twenty years [3-22]. The general objective is to find solutions to improve the accuracy of the correlation between IMs and RF performances of

the circuit and to ensure a good robustness of the prediction during the production testing phase. Some techniques therefore seek to select the most relevant IMs, or even create new specific IMs with new stimuli or power supply values, or even integrate additional sensors into the circuit. Other studies seek to find the most relevant machine-learning algorithms, with reproducible and generic strategies to optimize the accuracy and robustness of the estimations.

In this paper, we focus on a new kind of learning algorithms, namely ensemble methods, to see whether they can improve the indirect test efficiency. Indeed, in the recent years, ensemble methods have gained in popularity and have shown their superiority over classical learning algorithms in a number of application domains. However, to the best of our knowledge, only a limited number of works have addressed the use of these methods in the specific context of analog/RF indirect test. An ensemble method is used in [9] to implement indirect test on a LNA circuit, but no comparison with classical methods is realized, neither with different types of ensemble methods. A study is presented in [23] that gives a first analysis of the benefit that can be achieved by using different types of ensemble methods, in terms of model accuracy and reliability. In this paper, we extend this latter work with a more detailed analysis with respect to the different metrics; we also explore the influence of the training set size and investigate the potentialities of a two-tier test flow.

This paper is organized as follows. Section 2 recalls the basics of the indirect test approach. Section 3 gives an overview of the classical methods commonly used to build a regression model and introduces the principle of three types of ensemble methods. The experimental protocol developed to perform the comparative analysis between classical and ensemble methods, and the practical case study under investigation are presented in Section 4. First results are discussed in section 5 regarding the quality of the generated models, for each RF performance to be evaluated. Section 6 is devoted to the influence of the training set size. A global analysis of the results is then presented in Section 7 and the efficiency of classical and ensemble models is compared in the context of a two-tier test flow. Finally, Section 8 concludes the paper.

2 Indirect Test Principle

The underlying idea of indirect testing is that process variations that affect the device performances also affect non-conventional low-cost indirect parameters. If the correlation between the indirect parameter space and the specification space can be established, then specifications may be verified using only the low-cost indirect signatures. Unfortunately, the relation between these two sets of parameters is complex and cannot be simply identified with an analytic function. The solution commonly implemented uses machine-learning algorithms.

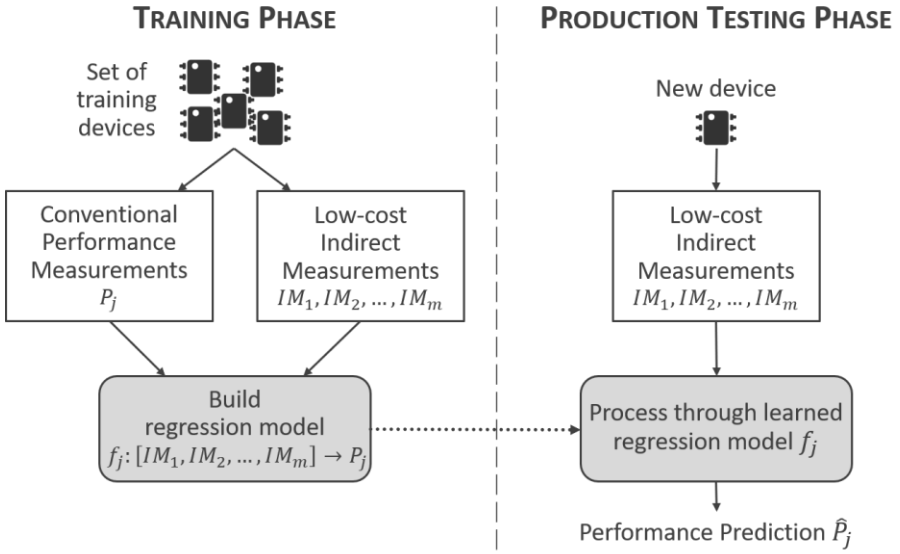

Fig.1. Indirect test synopsis

The indirect test synopsis is actually split into two distinct phases, namely training and production testing phases, as illustrated in Figure 1. The idea is to learn during the training phase the unknown dependency between the low-cost indirect measurements (IM_i) and the conventional performance measurements (P_j). To achieve this, both the

specification tests and the low-cost measurements are performed on a set of training devices and a machine-learning algorithm is trained to build regression models that map the indirect parameter space to the performance parameter space. During the production testing phase, only the low-cost indirect measurements are performed and the specifications of every new device are predicted using the mapping learned in the initial training phase.

3 Regression Models: Overview

3.1 Classical Methods

The classical approach to predict the value of a target parameter on unseen instances is to build a single regression model. Many different algorithms exist to perform this task. The most popular algorithms used in the context of indirect test are Multiple Linear Regression (MLR), Multi-Adaptive Regression Splines (MARS), and Support Vector Machine (SVM). The fundamentals of these models are briefly described hereafter.

3.1.1 Multiple Linear Regression (MLR)

An MLR model is a simple analytical model that expresses a linear relationship between the output variable (the circuit performance to be predicted) and multiple individual input variables (the indirect measurements). The main interest of this model is that it gives a clear idea of how the inputs affect the output. Moreover, because of its extreme simplicity, it is very fast to compute. However, because it assumes only linear relationship between the input and output variables, it might not be appropriate to correctly represent complex data.

3.1.2 Multi-Adpatative Regression Splines (MARS)

A more refined model is a MARS model, which is based on non-parametric regression. It can be considered as an extension of linear models. Nonetheless, it includes automatic modeling of nonlinearities and interactions between variables. In particular, the technique involves the partitioning of the input space into several regions, each one with its own regression equation. The algorithm automatically computes the different parameters related to the partitioning of the input space and the combination of the variables. The main advantage of a MARS model is that it makes no assumption about the underlying functional relationship between the dependent and independent variables. In counterpart, its computational cost is much higher than for MLR models.

3.1.3 Support Vector Machine (SVM)

SVM is a supervised machine learning algorithm which can be used for both classification or regression challenges. The general objective of the SVM algorithm is to find a hyperplane which separates the data into classes. The algorithm exploits built-in kernel methods that transform non-linear data into an optimized linear representation in a higher dimensional space. It is capable of solving linear and non-linear problems. Support Vector Machine has been used in several different domains and has demonstrated its ability to produce accurate results with reasonable computational power.

3.2 Ensemble Methods

When it comes to choosing a prediction model, there is no obvious winner among the various proposed algorithms; each model has its shortfalls and advantages. To cope with the model performance dependency on the size and the structure of the training data, researchers have started to use multiple regression models and aggregate their outcomes to get the final prediction results. The idea is that with an appropriate combination of diverse individual models, it should be possible to exploit the strengths and overcome the weaknesses of the individual models and obtain better overall predictive performance. This approach is called ensemble learning, which refers to the procedures used to train multiple individual regression models (base learners) and combine their outputs in order to improve the stability and the predictive power of the ensemble model. Numerous methods for constructing ensemble models have been proposed in the literature [24], which include parallel and sequential methods, based either on a single type of base learners (homogenous ensemble model) or learners of different types (heterogeneous ensemble model). In this section, we describe the general principle of the three most popular methods.

3.2.1 Bagging

Bagging stands for bootstrap aggregation. The basic motivation for bagging is to decrease the variance by averaging multiple estimates. The principle consists in using bootstrap resampling (random sampling with replacement) to generate different data subsets from the original training set. Multiple base learners are then trained on these random

subsets and the outputs of the base learners are averaged to produce the final estimate. Bagging is a parallel ensemble method that can be applied with any type of prediction model, but the most common application is with decision trees. A very popular algorithm that follows the bagging technique is Random Forest (RandF), which uses decision trees as base learners but also randomizes the trees by selecting a random subset of features (Indirect Measurements in our context).

3.2.2 Boosting

Boosting is also a method that relies on building multiple base learners on different datasets. However, unlike bagging, boosting is a sequential method. The idea is to incrementally build an ensemble by training, at each iteration, a predictor model that will correct its predecessor, by focusing on the under-fitted samples that present a large prediction error. The most popular method of boosting is AdaBoost (Adaptive Boosting). In this technique, the first predictor is learned on the entire dataset, with an equal weight assigned to all training samples. Then at each iteration, the algorithm modifies the weights of the training samples, giving higher weights to under-fitted samples. Finally, results of all predictors are aggregated using a weighting sum to produce the final prediction. Another popular boosting technique is Gradient Boosting. As in the AdaBoost algorithm, a new model is generated at each iteration with the objective to correct the predecessor model; the main difference is that the algorithm tries to fit residual errors made by the previous predictor instead of updating the training samples weights. As for bagging, boosting techniques can be applied with any type of prediction model, but they are usually applied with decision tree methods.

3.2.3 Stacking method

Stacking is a heterogeneous ensemble method that exploits a different principle than bagging and boosting techniques, that is based on the concept of a meta learner. The main concept is to use a prediction model to perform the aggregation of multiple base models. Practically, the technique involves two phases. First, multiple base learners are trained on the same dataset, generally using models of different types. The outputs of these base learners are then used to train a higher-level learner, called meta-learner. The two essential differences between stacking and bagging/boosting are: (i) the base models are not obtained by manipulating the training data but by using different model types, and (ii) the aggregation of the different base models is not performed by a simple combiner such as averaging or weighted sum but by a prediction model.

4 Experimental setup

4.1 Protocol of Experiments

In order to explore whether ensemble methods can bring benefits over classical methods, the experimental protocol depicted in Figure 2 has been defined. It involves 4 main phases that consist in (i) population partitioning, (ii) feature selection, (iii) model construction and (iv) test efficiency evaluation. Details on these different phases are given hereafter.

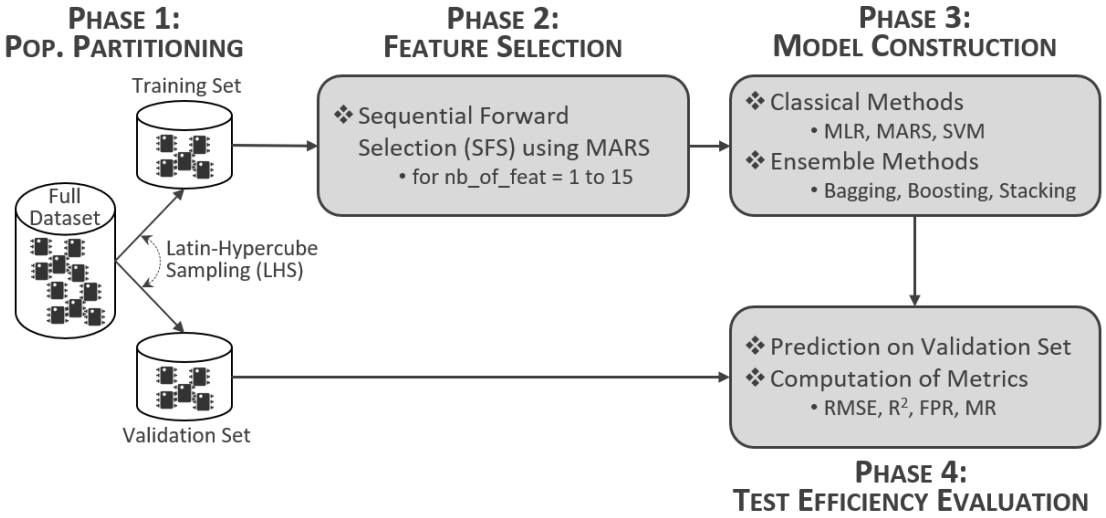

Fig.2. General overview of the experimental protocol

The first phase involves the partitioning of the population into two different sets. The first one will be used to train the prediction model and the second one with be used to evaluate the constructed model. Note that it is important to evaluate the performance of the model on different instances than the ones used for training, to verify the generalization ability of the model and avoid issues related to overfitting. In this work, we use Latin Hypercube Sampling (LHS) to perform the partitioning. This technique ensures that both the training and validation sets have similar statistical characteristics.

The second phase consists in selecting pertinent *IMs* among the set of available measurements. This problem of selecting a subset of features among a larger set is a recurrent problem in the field of machine-learning, known as feature selection. Various algorithms have been proposed, which can be divided into three categories, namely filters, wrappers and embedded methods [25]. In the context of indirect testing, the solution commonly employed is a wrapper method based on Sequential Forward Selection (SFS). The procedure starts by building a regression model for each available *IM* and selecting the *IM* that generates the model with the minimum prediction error (lowest Root Mean Square Error score). At the second iteration, a regression model is built for each pair of *IMs* that includes the previously selected *IM*; the pair that gives the best model is then selected. The process then continues with triplets and so on, until a stopping criterion is reached, for instance the number of selected *IMs* reaches a maximum target limit. In this work, we have implemented such a procedure using the MARS algorithm to build the regression models and limiting the search to a maximum of 15 features. The IM selection through SFS procedure is performed independently for each RF specification. The set of selected IMs is different from one RF specification to the other, however some IMs appear in the selected set of several specifications.

The third phase consists in building a regression model using the features selected in the previous phase. In this work, the objective is to investigate whether ensemble methods can outperform classical regression methods. We have chosen to compare three classical prediction models with five ensemble models. For the classical regression methods, we have implemented the three model types presented in section 3.1, namely MLR, MARS and SVM models. For the ensemble methods, we have implemented models belonging to the different categories presented in section 3.2:

- *Bagging*: one ensemble model is built from ten MARS models trained in parallel on ten bootstrap samples of the original training set.
- *Boosting*: one ensemble model is built using the AdaBoost algorithm with a sequential training of ten MARS models, and one ensemble model is built using the Gradient Boosting algorithm with 100 decision trees.
- *Stacking*: one ensemble model is built using the three classical models (MLR, MARS, SVM) as base models, and one ensemble model is built by adding a Random Forest (bagging algorithm applied on 300 decision trees) as a fourth base model. In both cases, the aggregation of the base learners is realized by the MARS algorithm.

Finally, the last phase concerns the evaluation of the test efficiency. In this phase, all the models built in the previous phase are used to achieve performance prediction of the devices in the validation set. Several metrics are then computed to evaluate the performance of the different prediction models.

The most commonly used metric to evaluate the quality of a model in the context of indirect testing is the Mean Square Error (*MSE*) or the Root Mean Square Error (*RMSE*), which is a measure of the difference between the values predicted by a model and the observed values. This metric gives information on the accuracy of a model. The interest of the *RMSE* score is that it is expressed in units of the variable of interest. It is computed as the square root of the average of squared errors:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (1)$$

where y_i is the actual performance value of the ith instance, $\hat{y}_i$ is the predicted performance value of the ith instance and n is the number of instances in the validation set.

Note that the *RMSE* score depends on the variable scale. Therefore, it can be used to compare different models for a given variable but not between different variables. To facilitate the comparison between variables with different scales, normalization can be applied. Although there is no consistent means of normalization in the literature, common choices are the mean or the range of the observed data. In this paper, we define the Normalized Root Mean Square Error (*NRMSE*), expressed in percentage, as the *RMSE* divided by the mean $\bar{y}$ of the observed data:

$$NRMSE = \frac{RMSE}{\bar{y}} \tag{2}$$

Another common metric used in statistics to evaluate the quality of a model is the coefficient of determination R^2, which is a measure of how well the regression predictions approximate the real data points. This score is a measure of the goodness-of-fit of a model and it is computed as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3}$$

The interest of the R^2 score is that it is a normalized score that ranges between 0 and 1 therefore permits comparison across different variables.

Note the *NRMSE* score can be related to the R^2 score with:

$$NRMSE = \frac{\sigma_y}{\bar{y}} * \sqrt{1 - R^2} = CV_y * \sqrt{1 - R^2} \tag{4}$$

where σ_y is the standard deviation of the observed data, and $CV_y = \sigma_y/\bar{y}$ is the coefficient of variation which corresponds to a standardized measure of the variability of the population.

This equation indicates that, despite normalization, the *NRMSE* score has a dependence with the observed data since it depends not only on the quality of the model through the R^2 score, but also on dispersion of the observed data through the coefficient of variation CV_y. Comparison of *NRMSE* scores between variables might be meaningless if the observed data for each variable present a very different dispersion. In contrast, the R^2 score permits fair comparison across different variables.

Another metric has been suggested in [16], which permits to quantify the prediction reliability of a model. This metric, called Failing Prediction Rate (*FPR*), expresses the percentage of circuits with a prediction error that exceeds the conventional measurement uncertainty ε_{meas}:

$$FPR = \frac{1}{n}\sum_{i=1}^{n}(|y_i - \hat{y}_i| > \varepsilon_{meas}) \quad \text{with} \quad \begin{matrix}(|y_i - \hat{y}_i| > \varepsilon_{meas}) = 1 & \text{if true} \\ (|y_i - \hat{y}_i| > \varepsilon_{meas}) = 0 & \text{otherwise}\end{matrix} \tag{5}$$

Lastly, if the test limits are available, we can compute another metric called the Misclassification Rate (*MR*). This metric simply expresses the ratio of misclassified circuits with respect to the total number of circuits.

In this paper, we will use all these metrics to present and comment results.

4.2 Case Study

The test vehicle is a Low-Noise Amplifier (LNA) for which we have production test data on 3,850 devices. More precisely, test data include the conventional measurements of three RF specification performances, namely the gain, the output power at 1dB compression point (P1dB) and the third-order intercept point (IP3). Test data also include 79 low-cost indirect measurements, which correspond to DC voltages on internal nodes (the device is equipped with an internal DC bus and internal DC probes) and DC signatures delivered by built-in process monitors. The distribution of the three RF performances is illustrated in Figure 3 and the main characteristics are summarized in Table I.

A first general comment is that the RF performances under investigation do not exhibit a Gaussian distribution. Another important point to highlight is that the three RF performances correspond to three different situations:

- For the gain, we observe a very tight distribution with dispersion of only 0.51% and a standard deviation that is even smaller than the measurement uncertainty. The test limits are located far away outside the distribution of available samples; as a consequence, there are no bad circuits with respect to the gain performance.
- For P1dB, we observe a slightly larger distribution with a dispersion around 1% and a standard deviation that is around twice the measurement uncertainty. For this performance, the lower test limit is located very close to the left tail of the distribution; three samples have a P1dB performance inferior to this limit, which means that only a negligible portion of the population (less than 0.1%) are bad circuits with respect to the P1dB performance.
- For IP3, we observe a significantly larger distribution with a dispersion around 2% but a standard deviation that is only about 1.5 times the measurement uncertainty. For this performance, the lower test limit falls within the distribution of available samples; 807 samples exhibit an IP3 performance inferior to this limit, which means that around 20% of the population are bad circuits with respect to the IP3 performance.

It will be particularly interesting to see how the indirect test approach, and more specifically the different types of model, are able to handle these three different situations. In this objective, the experimental protocol presented in the previous subsection has been applied on the three RF performances and results are commented in the following section.

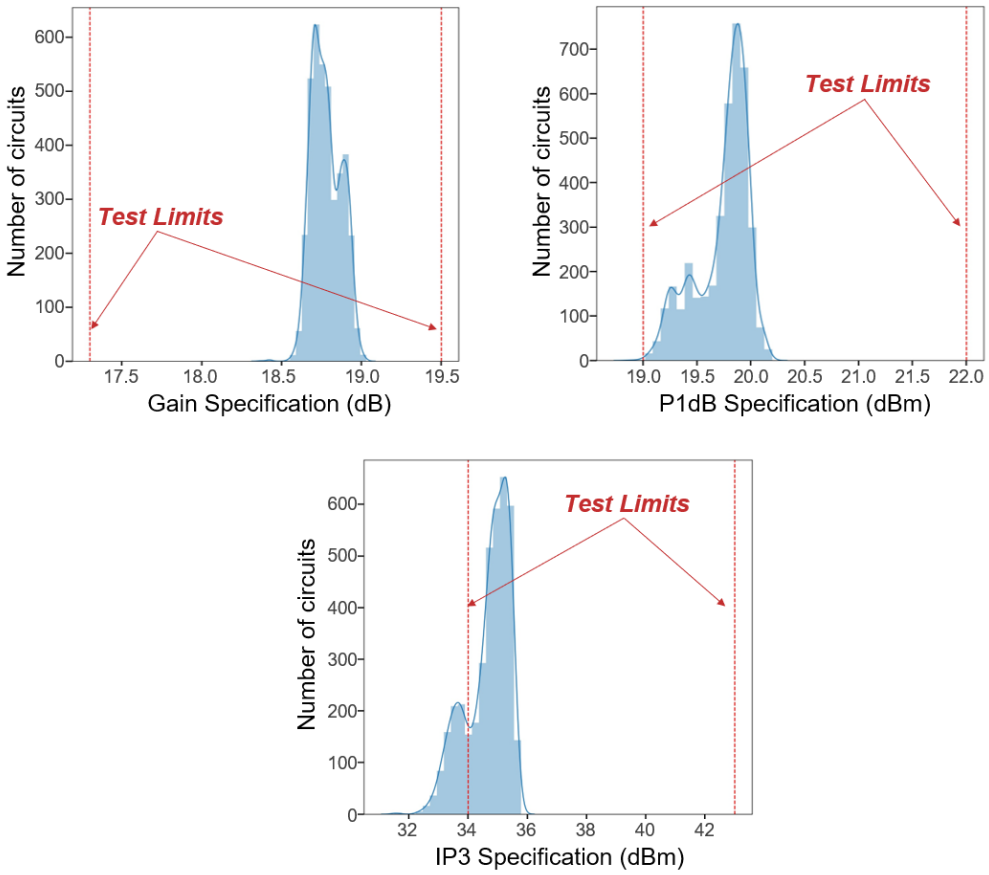

Fig.3. Distribution of the 3 RF performances under investigation

TABLE I. SUMMARY OF THE MAIN CHARACTERICTICS FOR THE 3 RF PERFORMANCES UNDER INVESTIGATION

	RF Performance		
	Gain	***P1dB***	***IP3***
Mean value	17.78dB	19.74dBm	34.68dBm
Standard deviation	0.09dB	0.24dBm	0.72dBm
Coefficient of variation	0.51%	1.22%	2.08%
Meas. uncertainty	0.1dB	0.1dB	0.5dB
Test limits	[17.3dB;19.5dB]	[19dBm;22dBm]	[34dBm;43dBm]
# good circuits	3850	3847	3043
# bad circuits	0	3	807

5 First results

5.1 Prediction of gain (G)

Figure 4 summarizes the comparison between classical and ensemble methods for the prediction of the gain specification. More precisely, it reports the evolution of R^2, *NRMSE* and *FPR* scores evaluated on the validation set with respect to the number of features used in the regression model for the different methods (classical models are plotted in dotted lines and ensemble models in solid lines).

Several comments arise from the analysis of these graphs. Regarding classical methods, there is a clear advantage to models generated by MARS algorithm compared to MLR and SVM. The best solution is actually obtained using MARS model built with nine features, with a R^2 score of 0.65, a *NRMSE* score of 0.29% and a *FPR* score of 2.9%. Regarding ensemble methods, models generated using stacking are more performing than models generated using boosting or bagging. The best solution corresponds to an ensemble model built with nine features that combines MLR, MARS, SVM and Random Forest (RandF) models. This model permits to reach a R^2 score of 0.72, a *NRMSE* score of 0.26% and a *FPR* score of 1.5%.

More generally for the gain specification, these results show that it is possible to obtain a benefit by using ensemble methods compared to classical methods, especially when stacking is applied. Compared to the best solution obtained using a classical method (MARS model in this case), the benefit is particularly visible on the achieved goodness-of-fit with a R^2 score that gains +0.07 and on the robustness with a *FPR* score that is reduced by a factor of almost two. The improvement is less visible on the accuracy with a *NRMSE* score that only reduces of 0.03%. However, it should be noticed that, whatever the method used to build the regression model and despite the fact that the R^2 score is relatively low, a very good accuracy is achieved for this specification. This good accuracy mainly comes from the fact that the observed population exhibits a very tight distribution with a very low coefficient of variation.

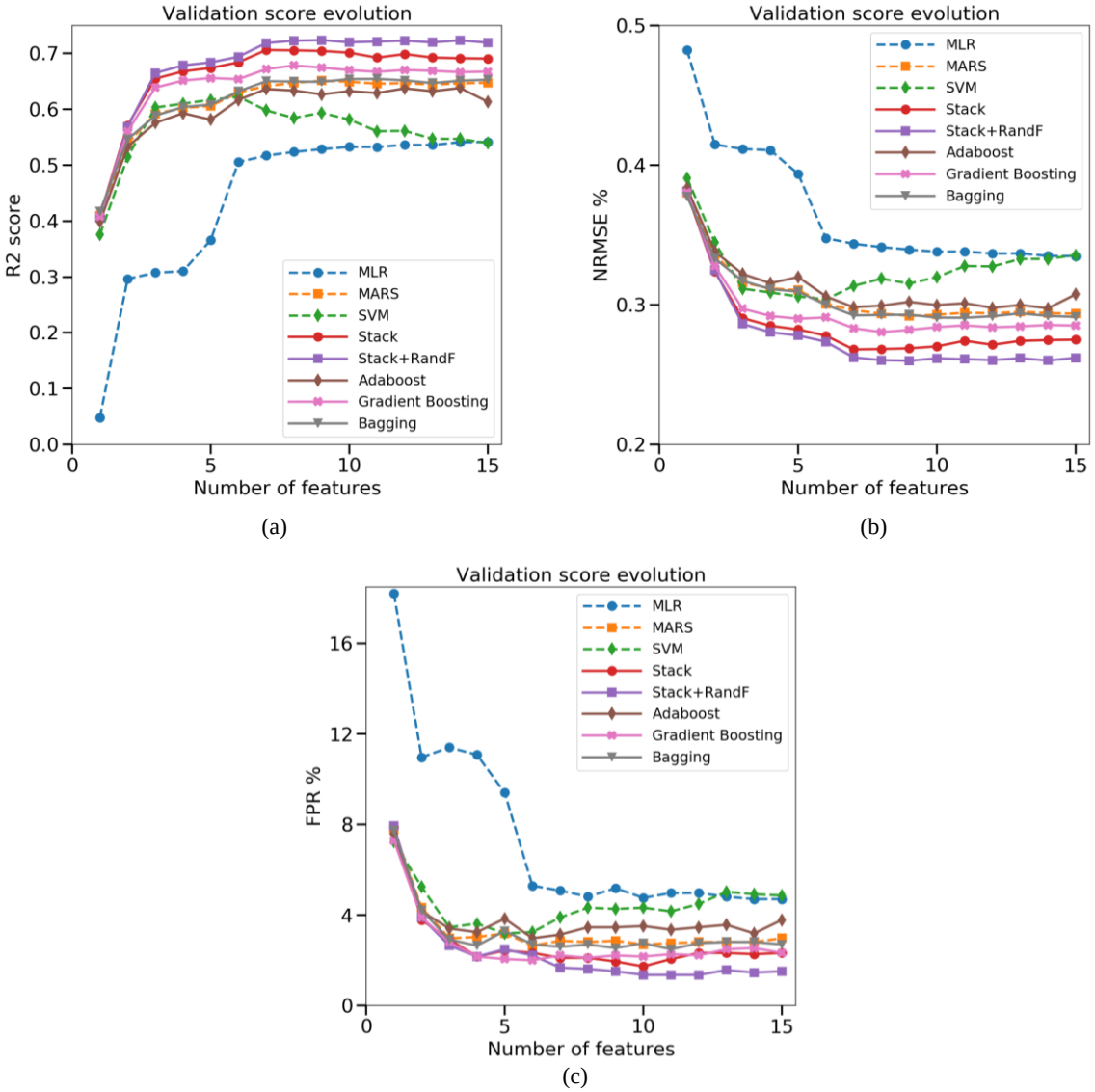

Fig.4. Comparison of classical and ensemble methods for gain prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score

5.2 Prediction of output power at 1dB compression point (P1dB)

Figure 5 summarizes the comparison between classical and ensemble methods for the prediction of the P1dB specification, in terms of R^2, *NRMSE* and *FPR* scores achieved on the validation set by using the different methods.

Regarding classical methods, unlike the gain specification, we can observe that SVM models are more powerful than MARS or MLR models, especially when only a limited number of features are used; results are then almost comparable when a higher number of features are used. The best solution is obtained using an SVM model built with eight features, with a R^2 score of 0.85, a *NRMSE* score of 0.48% and a *FPR* score of 12.3%. Regarding ensemble methods, we observe a similar trend than for the gain specification, i.e. models generated using stacking appear more powerful than models generated using boosting or bagging. The best solution corresponds to an ensemble model built with twelve features that combines MLR, MARS, SVM and Random Forest models. This model permits to reach a R^2 score of 0.87, a *NRMSE* score of 0.45% and a *FPR* score of 11.2%.

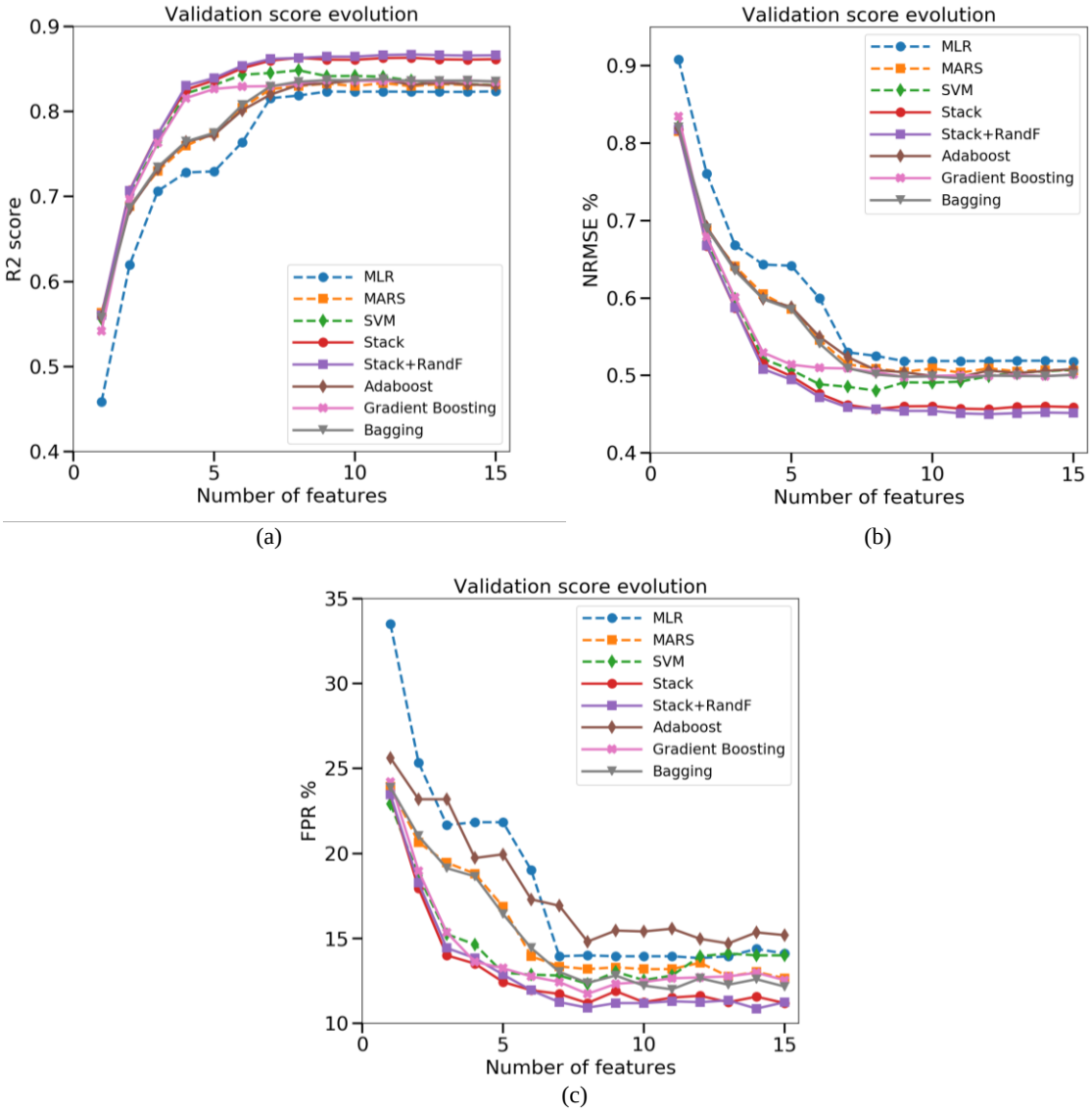

Fig.5. Comparison of classical and ensemble methods for P1dB prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score

Globally for the P1dB specification, there is a slight benefit in using ensemble models generated with stacking compared to the best model generated with a classical method (SVM model in this case), with a more limited improvement than for the gain specification. In this case, the R^2 score only gains +0.02 and both the *NRMSE* and *FPR*

scores remain in the same range. Notice that for this specification, despite the fact that the achieved goodness-of-fit is much better than for the gain, the achieved accuracy and the robustness are significantly lower than for the gain.

5.3 Prediction of third order intercept point (IP3)

Figure 6 summarizes the comparison between classical and ensemble methods for the prediction of the IP3 specification, in terms of R^2, *NRMSE* and *FPR* scores achieved on the validation set by using the different methods.

In case of the IP3 specification, a similar behavior than for the P1dB specification is observed, i.e. the more powerful models obtained with classical methods are SVM models and the more powerful models generated with ensemble methods are models generated with stacking. However, the benefit brought by the use of ensemble methods is not obvious in this case. Indeed, the best solution obtained with a classical method is a SVM model built with 14 features that exhibits a R^2 score of 0.93, a *NRMSE* score of 0.57% and a *FPR* score of 0.59%, while the best solution obtained with an ensemble method is a stacked model built with 14 features that exhibits a R^2 score of 0.94, a *NRMSE* score of 0.52% and a *FPR* score of 0.70%. There is therefore a small improvement of the R^2 and *NRMSE* scores but a small degradation of the *FPR* score. Notice that for this specification, whatever the method used to build the regression model, good results are obtained for all of the three metrics that express goodness-of-fit, accuracy and reliability

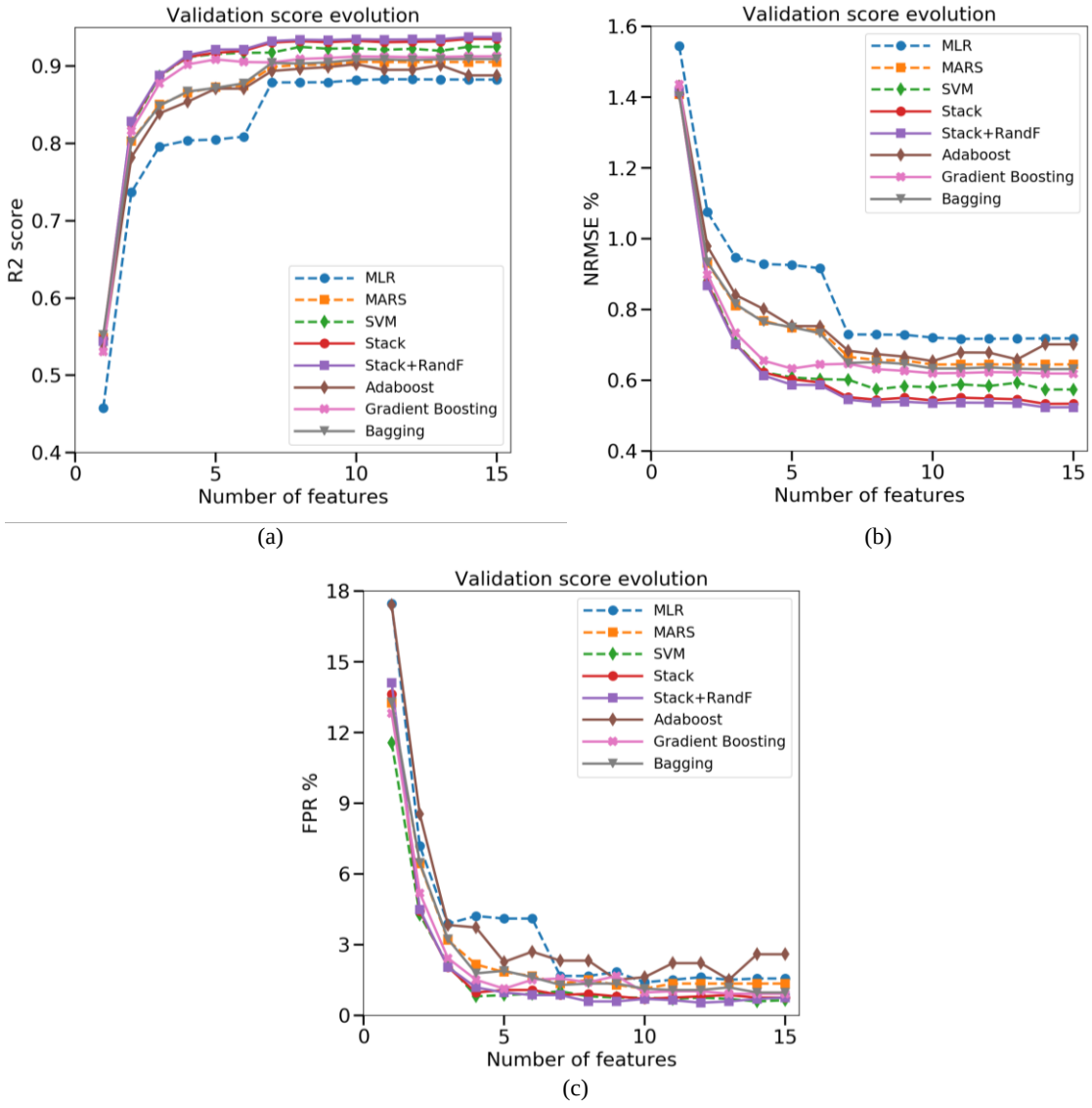

Fig.6. Comparison of classical and ensemble methods for IP3 prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score

6 Influence of the training set size

In this section, we want to investigate whether the use of ensemble learning might offer additional benefit with respect to the training set size. In this objective, we have performed additional experiments by reducing gradually the size of the training set. More precisely, we have considered three different sizes of training sets: 1,000, 500, and 200 circuits. The circuits for the various training sets have been chosen among the initial training set population of 2,000 circuits by using LHS, in order to preserve the distribution of each specification.

Then for each specification, we have selected the best ensemble model and the best classical model in terms of accuracy when the learning is performed on the initial training set of 2,000 circuits (MARS model for the Gain specification and SVM models for the P1dB and IP3 specifications in case of the classical approach, and stacked models for the three specifications in case of the ensemble approach). All these models have been trained on the different training sets and the R^2, *NRMSE* and *FPR* scores have been recorded. Note that whatever the size of the training set, all metrics are evaluated on the same validation set composed of 1,850 devices.

Results are summarized in Figure 7, which shows the evolution of the different metrics with respect to the size of the training set used, for the three specifications.

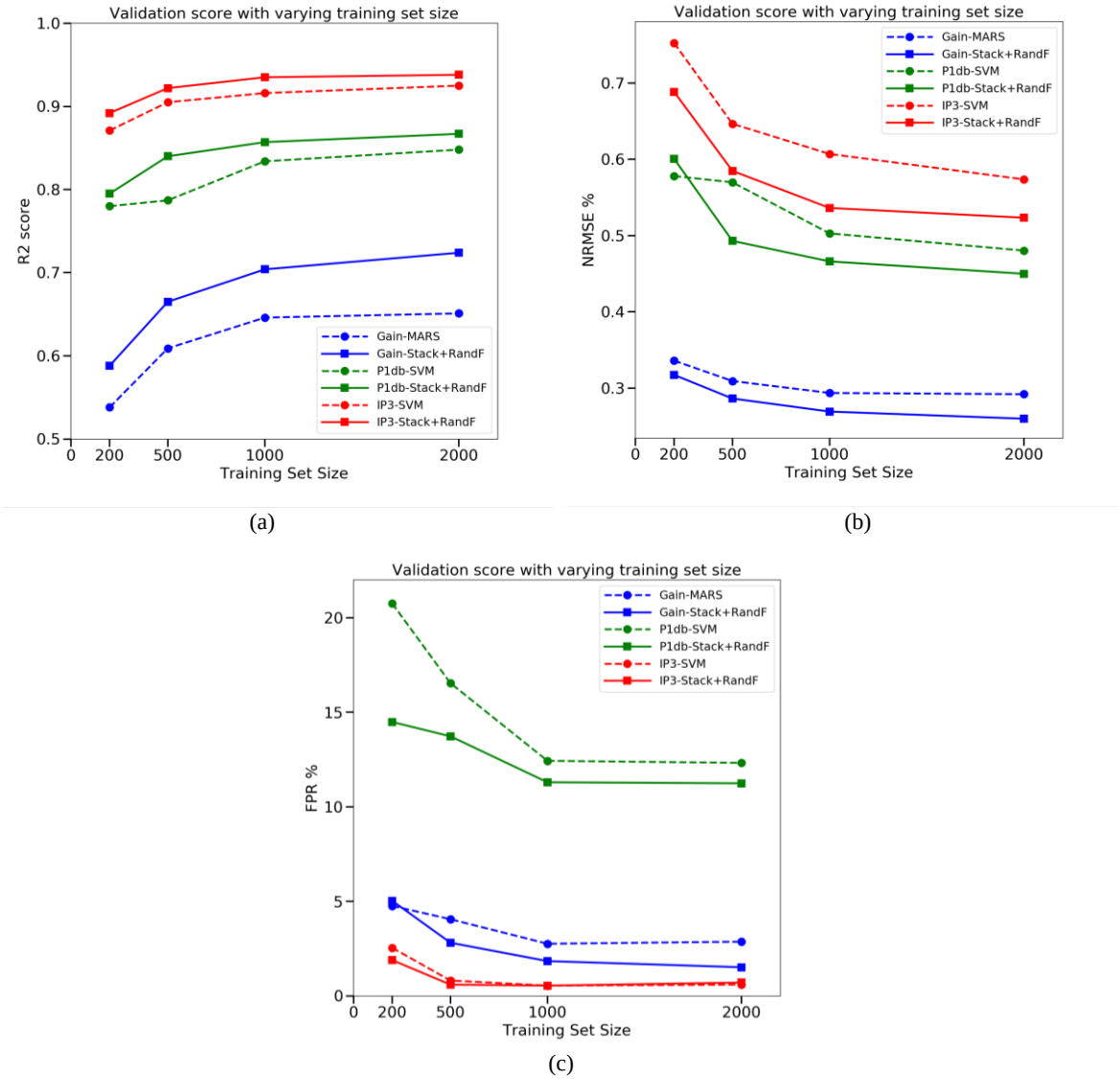

Fig.7. Influence of the training set size on performances achieved for the best classical and ensemble learning models for the 3 RF specifications: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score

As expected, there is a global degradation in the achieved scores as the training set size reduces, both for classical and ensemble models. This degradation is almost negligible when the training set sizes reduces from 2,000 to 1,000 devices, and then more pronounced when the training set size further reduces down to 500 and 200 devices. However, the situation differs depending on the considered metric and the evaluated specification.

In term of goodness-of-fit, the superiority of the ensemble model over the classical model is preserved whatever the size of the training set and whatever the evaluated specification. Nonetheless, we were expecting a more robust performance from ensemble learning, where we hoped to have a sort of stability while reducing the size of the training set, while we actually observe a decline in the R^2 score that is roughly similar to the case of classical algorithms.

In term of accuracy, we have the same trend overall, i.e. the ensemble models outperform the classical models for the different training set sizes and the three RF specifications, but the superiority of ensemble models does not increase as the training set size reduces. There is even an exception for the P1dB specification when the training set is composed only of 200 circuits. In this case, the *NRMSE* score achieved by the classical SVM model is slightly lower than the one achieved by the stacked ensemble model.

Finally, in term of reliability, the comparison between classical and ensemble models differs depending on the evaluated specification. For the IP3 specification, the best classical and ensemble models present a comparable performance with nearly equivalent *FPR* scores over the different training set sizes. For the Gain specification, there is a clear advantage for the best ensemble model compared to the best classic model when the learning is performed on 2,000 devices, but this advantage lessens as the training set size reduces and eventually vanishes when the learning is performed only on 200 devices. In contrast for the P1dB specification, the dominance of the ensemble model observed when the learning is performed on 2,000 devices increases as the size of the training set reduces. This is the only case where the use of the ensemble model leads to a better stability than the classical model.

Globally, this experiment shows that the benefit of using ensemble models is conserved, but it does not necessarily bring additional robustness with respect to the training set size.

7 Summary and discussion

Table II summarizes the best results obtained using either classical or ensemble methods for the three RF specifications, with learning performed on the training set of 2,000 devices. The criterion considered to select the "best" solution is the maximum value of R^2 score computed on the validation set. A first general comment is that the use of ensemble methods, and in particular ensemble methods based on stacking, permits to obtain an improvement in the goodness-of-fit of the generated model for the three specifications. However, the level of improvement is different in each case and seems to depend on the quality of the goodness-of-fit that can be reached by a single model. These results actually tend to indicate that the benefit of using the ensemble model reduces as the R^2 score reached by a single model increases. The use of ensemble methods also permits to obtain an improvement in the accuracy of the generated models for the three specifications, but it is a minor improvement with a reduction of the *NRMSE* score only of few hundredths of percentage point. In contrast, the situation is more diverse with respect to the reliability of the generated models. Indeed, we observe a significant reduction by about a factor of two of the *FPR* score in case of the gain specification, only a minor reduction of the *FPR* score in case of the P1dB specification, and a slight degradation of the *FPR* score in case of the IP3 specification.

Still, an important point to underline is that when using classical methods, the type of model that gives the best results differs depending on the specification (MARS or SVM). In contrast, ensemble models built with stacking always lead to the best results. It is an interesting characteristic to have a solution able to handle a variety of different situations. Furthermore, the use of ensemble learning especially stacking will not add any substantial implementation effort or raise the complexity of the procedure, since the difference in the computation time of training the various regression models is minimal, and in the order of seconds.

Hence globally, the use of ensemble models that are built using stacking appears to be an interesting option. Moreover, it should be mentioned that we didn't explore all the possibilities offered by stacking. Further improvements might be obtained, for instance by including other types of model as base learners which will add more diversity to the model collection, or by changing the type of the aggregating model (MARS model in this study).

TABLE II. COMPARISON BETWEEN CLASSICAL AND ENSEMBLE METHODS: SUMMARY OF BEST RESULTS FOR THE 3 RF PERFORMANCES

		Best solution selected from max(R^2) on validation set					
	RF Perf	***Model***	***R^2 (*)***	***NRMSE (*)***	***FPR (*)***	***MR (*)***	***# feat.***
Classical method	Gain	MARS	0.65	0.29%	2.86%	0%	9
	P1dB	SVM	0.85	0.48%	12.32%	0.1%	8
	IP3	SVM	0.93	0.57%	0.59%	4.2%	14
Ensemble method	Gain	Stack+RandF	0.72	0.26%	1.51%	0%	9
	P1dB	Stack+RandF	0.87	0.45%	11.24%	0.1%	12
	IP3	Stack+RandF	0.94	0.52%	0.70%	4.2%	14

(*) Score computed on validation set

More generally, this study also opens the question on what is a pertinent metric to evaluate indirect test efficiency. Indeed, results show that achieved performances significantly vary depending on the considered specification and the considered metric.

First, it appears that there is no evident relation between the goodness-of-fit, the accuracy and the reliability of a model. Indeed, for the gain specification, the best model has a rather low quality in terms of goodness-of-fit with a R^2 score around 0.7, but a good accuracy with a *NRMSE* below 0.3% and fairly good reliability with less than 2% of the devices that exhibit a prediction error which exceeds the classical measurement uncertainty. In contrast for the P1dB specification, we can obtain a reasonable quality in terms of goodness-of-fit with a R^2 score around 0.85 together with a good accuracy with a *NRMSE* smaller than 0.5%, but a relatively low reliability with more than 10% of the devices that exhibit a prediction error which exceeds the classical measurement uncertainty. Finally, for the IP3 specification, we can have at the same time good quality in terms of goodness-of-fit with a R^2 score higher than 0.9, good accuracy with a *NRMSE* around 0.5%, and good reliability with less than 1% of the devices that exhibit a prediction error that exceeds the classical measurement uncertainty.

Then, it should be highlighted that it is difficult to establish a link between these different metrics and the misclassification rate. Indeed, the misclassification rate strongly depends on the location of the test limits with respect to the distribution of available samples. For instance, in case of the gain specification, the test limits are located far away from the distribution; despite the relatively low goodness-of-fit of the models, all devices are correctly classified as good circuits and a perfect misclassification rate of 0% is achieved. In contrast for the IP3 specification, the lower test limit falls within the distribution; so even if we have models with very good goodness-of-fit, accuracy and reliability, around 4% of the circuits are misclassified, which can be considered as a non-negligible number. Yet, this result should be mitigated by the fact that all the misclassified circuits are located relatively close to the test limit, as illustrated in Figure 8, which highlights the location of misclassified circuits on the global IP3 distribution of the validation set. In fact, the computed misclassification rate might not be fully representative of the indirect test efficiency because it does not take into account the uncertainty that can affect the conventional measurement.

To further explain this point, let us analyze more in details Figure 8. When the measurement uncertainty is taken into account, it exists a region around the test limit where the circuits might be either good or bad circuits; only circuits outside this region can be trustfully defined as good or bad circuits by the conventional method. For our practical case on the IP3 specification, among the 1,850 circuits of the validation set, 400 are within the uncertainty region, 180 are trusted bad circuits and 1,270 are trusted good circuits. Now looking at the results of the indirect test, it appears that almost all the misclassified circuits are located within the uncertainty region, only five circuits being outside this region. The computed misclassification rate of 4% does not permit to reveal this situation. A more pertinent metric might be to compute the coverage of trusted classifications, i.e. the percentage of circuits that have a correct decision with the indirect prediction among the number of circuits that have a certain decision with the conventional measurement. For our case study, 1,450 circuits have a certain decision with the conventional measurement and 1,445 of them have a correct decision with the indirect prediction, which corresponds to a very good coverage of 99.66%. We believe that this metric can be more representative of the indirect test efficiency than the misclassification rate classically computed.

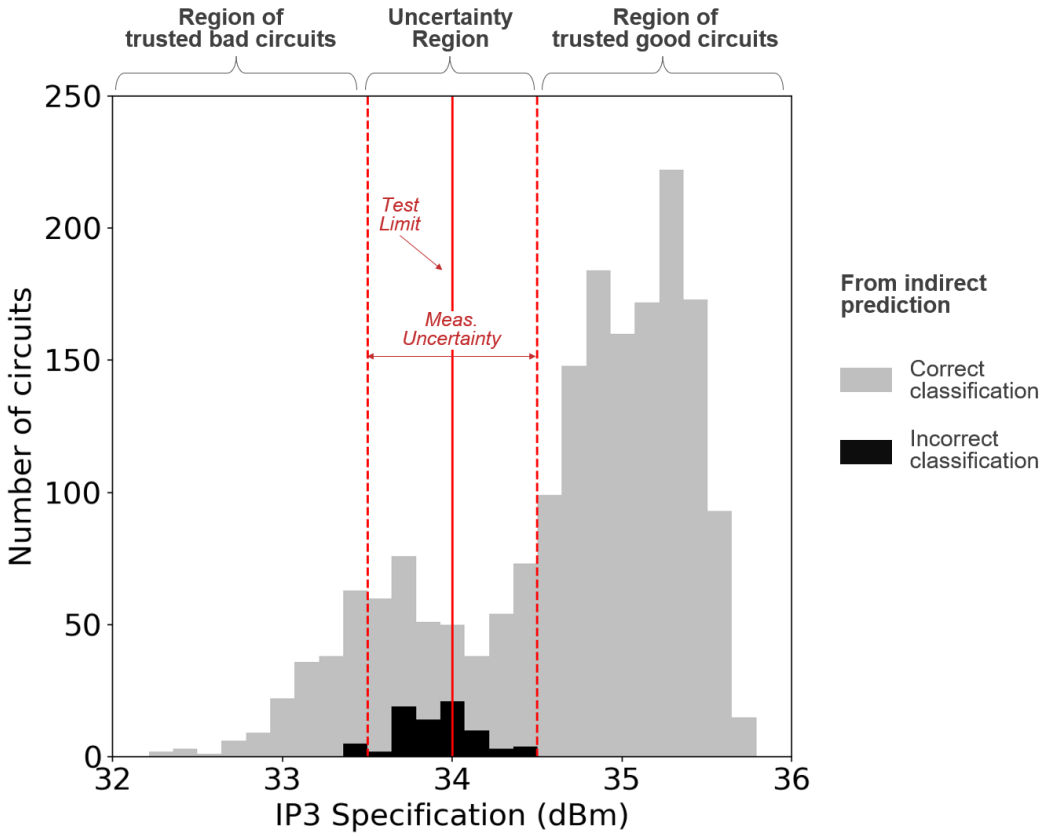

Fig.8. Illustration of misclassified devices by using the "Stack+RandF" ensemble model for IP3 specification

These observations pinpoint one of the main obstacles towards the wide deployment of indirect test in an industrial context, i.e. the difficulty to assess the confidence in the decision to classify a device as good or bad based only on indirect measurements. To tackle this issue, an interesting approach has been proposed in [6], which consists in implementing an adaptive test flow. The idea is to implement a two-tier test flow where the first tier corresponds to the indirect test and the second tier corresponds to the conventional specification test. The principle is that during production test, every device is first processed by the indirect test. If the confidence in the decision proposed by this first tier is high enough, the device is labeled according to the indirect test decision; otherwise it goes to the second tier where it is retested through the standard specification test. The objective of this approach is to preserve the test quality of specification testing while leveraging the low-cost of indirect testing. This solution has been investigated in the context of classification-oriented indirect test, where the machine-learning algorithm is not trained to predict the device performances but to directly perform classification in the indirect measurements space. The solution that has been developed relies on the allocation of appropriate guard-bands in the indirect measurements space in order to assess whether the device is suspect to misclassification or whether the indirect test decision can be trusted with good confidence.

In our case the situation is different because we are in the context of prediction-oriented indirect test, but the principle of a two-tier test flow can be preserved. Based on the observation that almost all the misclassified circuits are located relatively close to the test limit and that very good coverage is obtained outside the uncertainty region, the idea is to define a tolerance zone around the test limit. Every device with a prediction that falls outside this tolerance zone will be classified as a good or bad device based on the estimated performance, while every device with a prediction that falls within the tolerance zone will be directed to the second tier to be evaluated through conventional specification test. The synoptic of this two-tier test flow is illustrated in Figure 9; the training phase remains unchanged, only the production testing phase is modified.

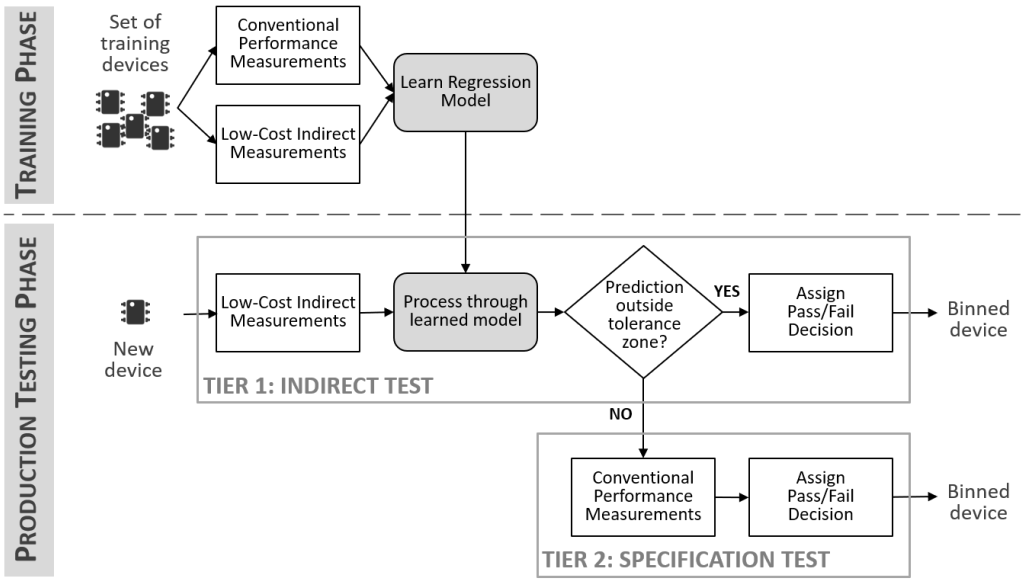

Fig.9. Synopsis of the adaptive two-tier test flow

The first natural solution to choose the size of the tolerance zone is to consider the conventional measurement uncertainty. However, an interesting characteristic of this two-tier test flow is that by varying the size of the tolerance zone around the test limit, we can have an exploration of the tradeoff between test quality and test cost, and therefore facilitates the development of cost-effective test plan. Indeed, in the initial implementation of the indirect test, there is no tolerance zone around the test limit and 100% of the devices evaluated during production test are processed by the low-cost first tier. The test cost is therefore minimum but the test quality expressed in terms of misclassification rate might not be sufficient to meet with industrial constraints. By creating and enlarging the tolerance zone around the test limit, we can expect an improvement of the test quality with a decrease of the misclassification rate, but at the expense of retesting a number of devices and therefore lessening the benefit in terms of test cost reduction.

This potentiality has been investigated on our test vehicle with respect to the IP3 specification. In particular, we have compared the tradeoff between the global misclassification rate of the test flow and the number of devices that need to be retested with a conventional specification test, for the best performing classical and ensemble models. Results are summarized in Figure 10.

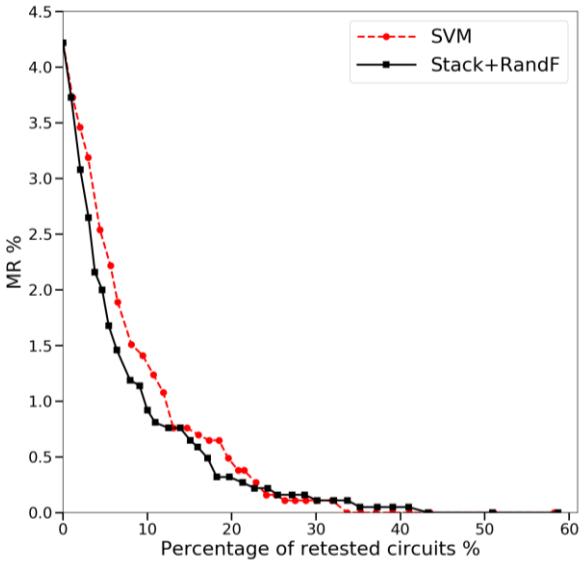

Fig.10. Tradeoff curve between the misclassification rate and the percentage of retested device, for the best classical and ensemble learning models

The analysis of these results reveals that the two-tier test flow offers the possibility to reduce the global testing costs without compromising the test quality. Indeed, this strategy allows to achieve the same quality as a conventional specification test with a perfect 0% misclassification rate, but only 33% of the devices that need to be evaluated by the expensive conventional specification test while the other 67% are correctly evaluated by the low-cost indirect test. Moreover, these results indicate that further benefit can be obtained if a certain level of acceptance can be tolerated on the misclassification rate. Indeed, starting from the initial point where all the devices are evaluated only by the indirect test flow, which gives a misclassification rate around 4%, there is a rapid decrease in the achieved misclassification rate when conventionally re-testing a part of the circuits in a second tier. For instance, a misclassification rate below 1% can be obtained with less than 10% of the devices that need to go through a specification test, leading to a substantial benefit in terms of testing costs compared to the conventional specification test of all circuits. Here again, it is important to underline that the 1% of misclassified devices are not fully defective devices, but just devices that marginally fail the specification. Finally, regarding the comparison between the best classical model and the best ensemble model, there is no massive difference but the ensemble model seems a bit more performing, especially when a level of acceptance can be tolerated on the misclassification rate.

8 Conclusion

This paper reports an investigation on the benefit that could be provided by using ensemble methods for indirect testing of RF circuits. Different ensemble methods based on bagging, boosting and stacking have been studied and compared to classical individual models, namely MLR, MARS and SVM models, in terms of goodness-of-fit, accuracy and reliability. The experimental protocol that has been developed for this purpose has been applied to the practical case of study of a Low-Noise Amplifier for which we have production test data related to three RF performances and several dozens of low-cost indirect measurements. From this study, it appears that ensemble models built with stacking lead to the best models compared to ensemble models built with bagging or boosting in all cases (RF performance, number of features, training set size). Furthermore, this study shows that, in most situations, such models surpass classical individual models' performance, both in terms of accuracy and reliability, and tend to have a stronger predictive power. Overall, ensemble models built with stacking appear to be the most suitable solution for a wide range of situations. This study should be deepened by exploring and adding more diversity to the model collection (i.e. including other types of model as base learners), or by changing the type of the aggregating model (MARS model in this study). Finally, this paper highlights a meaningful question in the context of indirect RF testing on the pertinence of the metrics used to qualify a model. Not only are the metrics of goodness-of-fit, accuracy, and reliability independent of each other, but also it is difficult to relate them to an industrial test misclassification rate. This paper then proposes a new way of computing the misclassification rate by taking into account the conventional measurement uncertainty: the coverage of trusted classifications. In the extension of this idea, a two-step test flow is proposed that allows choosing the best compromise between test cost and test quality for each type of device and specification. This flow consists in re-testing by conventional measurement of RF specifications only those circuits for which the level of confidence in the classification decision by indirect testing is not satisfactory, offering the possibility to reduce the global testing costs without compromising the test quality.

Acknowledgment

This work has been carried out under the framework of PENTA-EUREKA project "HADES: Hierarchy-Aware and secure embedded test infrastructure for Dependability and performance Enhancement of integrated Systems".

References

[1] V.G. Gil, A.J. Gines Arteaga, G. Léger, "Assessing AMS-RF Test Quality by Defect Simulation", IEEE Transactions on Device and Materials Reliability, vol. 19, no. 1, pp. 55-63, 2019.

[2] J. Carballido et al., "A Programmable Calibration/BIST Engine for RF and Analog Blocks in SoCs Integrated in a 32 nm CMOS WiFi Transceiver", IEEE Journal of Solid-State Circuits, vol. 48, no. 7, pp. 1669-1679, 2013.

[3] P.N. Variyam, A. Chatterjee, “Enhancing test effectiveness for analog circuits using synthesized measurements”, Proc. IEEE VLSI Test Symposium (VTS), pp. 132-137, 1998.

[4] P. N. Variyam et al., “Prediction of analog performance parameters using fast transient testing,” IEEE Trans On Computer-Aided Design of Integrated Circuits and Systems, Vol. 21, no. 3, pp. 349–361, 2002.

[5] S. Ellouz et al., “Combining internal probing with artficial neural networks for optimal RFIC testing”, Proc. IEEE International Test Conference (ITC), p.9, 2006.

[6] H. Stratigopoulos and Y. Makris, “Error Moderation in Low-Cost Machine-Learning-Based Analog/RF Testing”, IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 27, no. 2, pp. 339-351, 2008.

[7] H.-G. Stratigopoulos et al., “Enrichment of limited training sets in machine-learning-based analog/RF test”, Proc. Design Automation Test Conference (DATE), pp. 1668–1673, 2009.

[8] L. Abdallah et al., “Sensors for built-in alternate RF test,” Proc. IEEE European Test Symposium (ETS), pp. 49-54, 2010.

[9] M.J. Barragan, et al., “Improving the Accuracy of RF Alternate Test Using Multi-VDD Conditions: Application to Envelope-Based Test of LNAs”, Proc. IEEE Asian Test Symposium (ATS), pp. 359-364, 2011.

[10] H. Ayari et al., “Smart selection of indirect parameters for DC-based alternate RF IC testing”, Proc. IEEE VLSI Test Symposium (VTS), pp. 19-24, 2012.

[11] H. Ayari, et al., “Making predictive analog/RF alternate test strategy independent of training set size”, Proc. IEEE International Test Conference (ITC), p.9, 2012.

[12] H. Stratigopoulos and S. Mir, “Adaptive Alternate Analog Test”, IEEE Design & Test of Computers, Vol. 29, no. 4, pp. 71-79, 2012.

[13] M.J. Barragan, G. Leger, “Efficient selection of signatures for analog/RF alternate test”, Proc. European Test Symposium (ETS), p.6, 2013.

[14] S. Larguech et al., “Efficiency evaluation of analog/RF alternate test: Comparative study of indirect measurement selection strategies”, Microelectronics Journal, Vol. 46, Issue 11, pp. 1091-1102, 2015.

[15] A. Dimakos, et al., “Test and Calibration of RF Circuits Using Built-in Non-intrusive Sensors”, Proc. IEEE Computer Society Annual Symp. on VLSI (ISVLSI), pp. 627-627, 2015.

[16] S. Larguech, et al., “A Framework for Efficient Implementation of Analog/RF Alternate Test with Model Redundancy”, Proc. IEEE Computer Society Annual Symp. on VLSI (ISVLSI), pp. 621-626, 2015.

[17] M. J. Barragan and G. Leger, “A Procedure for Alternate Test Feature Design and Selection”, IEEE Design & Test, Vol. 32, no. 1, pp. 18-25, 2015.

[18] H. Stratigopoulos, “Machine learning applications in IC testing”, Proc. IEEE European Test Symposium (ETS), p.10, 2018.

[19] H. El Badawi et al., “Which metrics to use for RF indirect test strategy?”, Proc. International Conf. on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), p.4, 2019.

[20] M. J. Barragan et al., “On the use of causal feature selection in the context of machine-learning indirect test”, Proc. Design Automation Test Conference (DATE), pp. 276-279, 2019.

[21] A. Dimakos et al., "Parametric Built-In Test for 65nm RF LNA Using Non-Intrusive Variation-Aware Sensors", Journal of Electronic Testing, Vol. 31, pp. 381-394, 2015.

[22] P. Kansara et al., "Dynamic Analog/RF Alternate Test Strategies Based on On-chip Learning », Journal of Electronic Testing, Vol. 34, pp. 337-349, 2018.

[23] H. El Badawi et al., “Use of ensemble methods for indirect test of RF circuits: can it bring benefits?”, Proc. IEEE Latin American Test Symposium (LATS), pp. 1-6, 2019.

[24] Z-H. Zhou, “Ensemble Methods: Foundations and Algorithms”, Chapman and Hall/CRC, Machine Learning & Pattern Recognition Series, 1st Edition, 2012.

[25] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection”, Journal of Machine Learning Research, Vol. 3, pp. 1157-1182, 2003.