Methodologie pour une mise en oeuvre efficace du test indirect pour circuits intégrés analogique/RF

Hassan El Badawi

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Génie Informatique, Automatique et Traitement du Signal

École doctorale Information, Structures, Systèmes

Unité de recherche Laboratoire d'informatique, de Robotique et de Microélectronique de Montpellier (LIRMM)

Methodology for Efficient Implementation of Indirect Testing for Analog/RF Integrated Circuits

Présentée par Hassan EL BADAWI
Le 26 Novembre 2020

Sous la direction de Serge BERNARD et Florence AZAIS

Devant le jury composé de

M. Gildas LEGER	Professeur, IMSE-CNM	Rapporteur
M. Haralampos STRATIGOPOULOS	Directeur de recherche, CNRS-LIP6	Rapporteur
M. Michel RENOVELL	Directeur de recherche CNRS-LIRMM	Président du jury
M. François LEFEVRE	Ingénieur, NXP Semiconductors	Examinateur
Mme. Florence AZAIS	Chargé de Recherche, CNRS-LIRMM	Co-directrice de thèse
M. Serge BERNARD	Directeur de recherche, CNRS-LIRMM	Directeur de thèse
Mme. Mariane COMTE	Maître de conférences, Polytech/ Univ. Montpellier	Invitée
M. Vincent KERZERHO	Chargé de Recherche, CNRS-LIRMM	Invité

UNIVERSITÉ DE MONTPELLIER

Contents

Contents iii

List of Figures vi

List of Tables ix

Acknowledgments x

Résumé xi

Introduction 1

1 Indirect Test for Analog/RF IC 3
- 1.1 Introduction . . . 3
- 1.2 Concept of an Indirect Test Strategy . . . 4
  - 1.2.1 Classification-oriented Indirect Test . . . 4
  - 1.2.2 Prediction-oriented Indirect Test . . . 5
- 1.3 Synopsis of an Indirect Test Strategy . . . 5
- 1.4 Classical Regression Algorithms . . . 6
  - 1.4.1 Multiple Linear Regression . . . 6
  - 1.4.2 Multivariate Adaptive Regression Splines . . . 7
  - 1.4.3 Support Vector Machines . . . 8
  - 1.4.4 Decision Trees . . . 9
- 1.5 Indirect Measurement Selection . . . 10
  - 1.5.1 Filter Methods . . . 12
  - 1.5.2 Wrapper Methods . . . 14
  - 1.5.3 Embedded Methods . . . 15
  - 1.5.4 Hybrid Methods . . . 16
- 1.6 Evaluation Metrics . . . 16
  - 1.6.1 Normalized Root Mean Square Error . . . 16
  - 1.6.2 Coefficient of Determination . . . 17
  - 1.6.3 Failing Prediction Rate . . . 17
  - 1.6.4 Misclassification Rate . . . 17
- 1.7 Limitations . . . 18

1.8 Conclusions 19

2 Ensemble Learning **20**
2.1 Introduction 20
2.2 Ensemble Learning 20
2.2.1 Bagging 21
2.2.2 Boosting 22
2.2.3 Stacking 25
2.3 Experimental Setup 26
2.3.1 Experimental Protocol 26
2.3.2 Case Study 29
2.3.3 Initial Results 31
2.3.4 Influence of the training set size 36
2.4 Results Summary 38
2.5 Conclusion 42

3 Adaptive Test Flow **44**
3.1 Introduction 44
3.2 Two-Tier Adaptive Test-Flow 44
3.2.1 Principle 44
3.2.2 State-of-the-art on adaptive indirect test 45
3.2.3 Proposed Solution 47
3.3 Experimental Protocol 49
3.4 Case Study 51
3.4.1 RF Product 51
3.4.2 Measurement Campaign 52
3.4.3 Data Preparation 53
3.5 Results 57
3.5.1 Model selection 57
3.5.2 Efficiency of classical indirect test implementation 60
3.5.3 Efficiency of two-tier adaptive test flow 65
3.6 Conclusion 68

4 Embedded Indirect Test for Performance Monitoring **69**
4.1 Introduction 69
4.2 Adaptation of the Indirect Test Strategy for Performance Monitoring . 70
4.2.1 Indirect Measurements 71
4.2.2 Digitization 71
4.2.3 Memory and Arithmetic 72
4.3 Case Study 73
4.3.1 Test vehicle: RF transceiver (NXP JN518x) 73
4.3.2 Dataset Collection: Measurement campaign 74
4.3.3 Dataset Analysis 76

4.4 Model Elaboration 82
4.4.1 Preliminary study: choice of model type 82
4.4.2 Implementation in the case study 84
4.5 Embedded Prediction 89
4.5.1 Theoretical study on ADC resolution 90
4.5.2 Implementation on the case study 91
4.6 Conclusion 98

Conclusion **99**

Bibliography **101**

Personal Publications **107**

List of Figures

1.1 Indirect test synopsis [11] 6
1.2 Simple Linear Regression 7
1.3 Hinge Function 8
1.4 Linear Support Vector Regression [14] 9
1.5 Decision Tree Decision Rules 9
1.6 Max Depth parameter effect 10

2.1 Principle of ensemble model construction with Bagging 22
2.2 Principle of ensemble model construction with AdaBoost 23
2.3 Principle of ensemble model construction with Gradient Boosting 24
2.4 Principle of ensemble model construction with Stacking 25
2.5 General overview of the experimental protocol 26
2.6 Distribution of the three RF specifications 29
2.7 Comparison of classical and ensemble methods for gain prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score 32
2.8 Comparison of classical and ensemble methods for P1dB prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score 33
2.9 Comparison of classical and ensemble methods for IP3 prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score 35
2.10 Influence of the training set size on performances achieved for the best classical and ensemble learning models for the 3 RF specifications: (a) R^2score, (b) *NRMSE* score and (c) *FPR* score 37
2.11 Illustration of misclassified devices by using the "Stack+RandF" ensemble model for IP3 specification 41

3.1 Two-Tier Adaptive test flow synopsis 45
3.2 Principle of confidence estimation in the proposed two-tier adaptive test flow. 48
3.3 Principle of confidence estimation in the proposed two-tier adaptive test flow. 49
3.4 Number of circuits identified by the filtering process vs. filter severity . . . 54
3.5 Representation of filtered circuits in terms of good and faulty circuits . . . 55
3.6 Repartition of filtered circuits over Train, Test and Validation subsets . . . 56
3.7 NRMSE score achieved on train and test sets for the different scenarios of learning population - MARS model 58

3.8 NRMSE score achieved on train and test sets for the different scenarios of learning population - SVM model 59
3.9 NRMSE and MR scores achieved on test and validation sets for the different scenarios of learning population - Tx-EVM 61
3.10 NRMSE and MR scores achieved on test and validation sets for the different scenarios of learning population - Tx-gain 62
3.11 NRMSE and MR scores achieved on test and validation sets for the different scenarios of learning population - Rx-gain 63
3.12 Trade-off curves between MR and percentage of retested devices - Tx-EVM 66
3.13 Trade-off curves between MR and percentage of retested devices - Tx-gain 66

4.1 On-line performance monitoring based on the indirect test strategy 70
4.2 High level hardware block diagram of the test vehicle 74
4.3 Measurement campaign for dataset collection 75
4.4 Histogram of the transmitted power level measurements for the 1024 observations 76
4.5 Boxplot of the power level measured under the 256 configurations for the four ICs 77
4.6 Boxplot of the DC values measured on the 1024 observations, for the eleven indirect measurements 78
4.7 Boxplot of the relative DC deviation measured on the 1024 observations, for the eleven indirect measurements 79
4.8 Boxplot of the mean relative DC deviation over the four ICs, for the eleven indirect measurements 79
4.9 Mean relative DC deviation over the 256 configurations, for the eleven indirect measurements 80
4.10 Histogram of DC30 measurements under the 256 configurations, for the four ICs 81
4.11 Histogram of DC35 measurements under the 256 configurations, for the four ICs 81
4.12 Comparison of MLR models constructed on the original IM space or on the enriched one 83
4.13 Predicted power level variation vs. measured power level variation 86
4.14 Normalized distribution of the prediction error 86
4.15 Principle of limit value determination and resulting detection range 88
4.16 Illustration of limit value and detection range for the case study 89
4.17 Influence of ADC resolution on prediction error on IC4 91
4.18 Illustration of the experimental setup 92
4.19 Flowchart of the on-line performance monitoring process 93
4.20 Embedded prediction of power level degradation on IC4 94
4.21 Boxplot of predicted values for power level degradation over the 1,000 re-sampling operation vs. the re-sample size, in the nominal and degraded configurations 96

4.22 Comparison of power level degradation prediction on IC4 using either ATE or embedded measurements 97

List of Tables

2.1 Main characteristics of the three RF performances 30
2.2 Statistical characteristics of the training and validation sets 31
2.3 Comparison between classical and ensemble methods: Summary of best results for the three RF performances 39

3.1 Characteristics of the full dataset for the three RF performances 52
3.2 Main characteristics of the learning and validation sets for the three RF performances 53
3.3 Characteristics of the filtered learning sets for the three RF performances . 57
3.4 Summary of the best results achieved under different scenarios of learning population for the three RF performances 60
3.5 Summary of results achieved by the two-tier adaptive test flow with an *MR* target of 0.1% for each RF performances 67

4.1 Training and Validation *RMSE* scores for the four possible learning combinations 85
4.2 Statistics of the prediction error on the 256 configurations 86
4.3 ADC measurement range used for each IM involved in the prediction model 90
4.4 Summary of prediction error on IC4 97

Acknowledgments

I would like to thank Dr. Haralampos Stratigopoulos, Dr. Gildas Leger and Dr. Michel Renovell for their implication. It is a pleasure and honor that they accepted to be part of my dissertation committee.

Whatever I write to thank the people who supervised and helped me to complete this work will not be sufficient. I would like to dearly thank Florence Azais, Serge Bernard, Mariane Comte, Vincent Kerzerho, and Francois Lefevre for their guidance and patience throughout the past three years, I will fondly remember this period of my life.

Résumé

Le test des circuits intégrés est une étape cruciale du processus de production car il permet de garantir la qualité des dispositifs fabriqués et le respect des spécifications. Lorsqu'il peut être réitéré *in situ* dans le cadre du fonctionnement des circuits, le test peut également permettre de vérifier leur fiabilité pendant leur cycle de vie. Dans le cas des blocs numériques, bien que leur complexité ait explosé au cours des dernières décennies, l'utilisation de techniques de test structurel avec une approche basée sur la détection de défauts a permis de limiter la part du coût de test dans le prix de revient de ces blocs. En revanche, pour les blocs analogiques et Radio-Fréquence (RF), bien qu'ils aient moins évolué en termes de complexité que les blocs numériques, leur coût de test a continué à prendre une part de plus en plus prépondérante dans leur coût de revient. La raison principale est qu'il n'existe aucun modèle de défaillance reconnu pour les blocs analogiques et RF et que les approches basées sur la détection de défauts ne sont donc pas adaptées. Par conséquent, les circuits analogiques et RF sont testés selon une approche fonctionnelle, qui repose sur la mesure des performances du circuit et la vérification de la conformité de ces performances vis-à-vis des spécifications garanties par le fabriquant. Cette approche garantit une qualité de test satisfaisante, mais les mesures requises nécessitent des équipements de test spécifiques très onéreux et des temps de test longs, ce qui induit des coûts de test extrêmement élevés. Par ailleurs, une problématique importante pour les circuits RF concerne le test au niveau "wafer". En effet, l'application et la capture de signaux RF doivent dans ce cas être réalisées par des équipements de test "sous pointes" ("probers"), ce qui s'avère délicat compte tenu de la présence d'éléments parasites dans la liaison équipement-circuit qui dégradent la qualité des signaux appliqués et capturés.

Dans ce contexte, il existe une forte demande pour le développement de solutions alternatives au test de spécifications afin de réduire le coût du test des circuits analogiques et RF. Une approche intéressante est l'utilisation d'une stratégie de test indirect basée sur l'utilisation d'algorithmes d'apprentissage automatique ("machine-learning"). Bien que proposée il y a plus de 10 ans, cette stratégie reste relativement peu utilisée à ce jour dans l'industrie, ce qui s'explique d'une part par sa complexité de mise en œuvre, qui nécessite de multiples choix dans diverses étapes de l'élaboration de la solution, et d'autre part par la difficulté à obtenir une évaluation fiable de son efficacité. L'objectif

de cette thèse est de répondre à ces différents enjeux en proposant une méthodologie qui permette d'explorer différentes options d'implantation et de guider l'ingénieur de test dans les choix à faire pour une implantation efficace.

Le premier chapitre de la thèse est consacré à poser les bases nécessaires à l'étude. Le principe de la stratégie de test indirect est tout d'abord présenté. Cette stratégie repose sur deux phases distinctes, à savoir la phase d'apprentissage et la phase de test de production. L'idée est d'établir, pendant la phase d'apprentissage, la dépendance inconnue entre des mesures indirectes pouvant être réalisées à faible coût (IM_i) et les mesures conventionnelles de performance (P_j). Pour cela, les mesures de performance et les mesures indirectes sont effectuées sur un ensemble de dispositifs d'apprentissage et un algorithme d'apprentissage automatique est utilisé pour construire des modèles de régression permettant de relier l'espace des paramètres indirects à l'espace des paramètres de performance. Par la suite, lors de la phase de test de production, seules les mesures indirectes sont effectuées et les performances de chaque nouveau circuit sont prédites à l'aide des modèles de régression construits lors de la phase initiale d'apprentissage. Cette stratégie peut être appliquée soit avec une approche dédiée à la classification directe des circuits, soit avec une approche dédiée à la prédiction des performances. Dans cette thèse, nous avons étudié une mise en œuvre dédiée à la prédiction des performances.

De nombreux éléments sont susceptibles d'affecter la qualité globale de la stratégie de test indirect. La deuxième partie du chapitre est consacrée à ces différents aspects, en s'appuyant sur les travaux proposés dans la littérature. Les modèles de régression classiques tels que la Régression Linéaire Multi-niveaux (MLR), la Régression Multivariée par Spline Adaptative (MARS), les Machines à Vecteurs de Support (SVM) et les Arbres de Décision (DT) sont ainsi présentés en détail. La sélection des mesures indirectes les plus pertinentes est un processus de recherche aussi important que le choix du modèle de prédiction. Les trois principaux types d'algorithmes de sélecetion des paramètres ("feature selection") sont donc analysés. Enfin, les métriques classiques utilisées pour évaluer et comparer différentes solutions sont définies, à savoir l'erreur quadratique moyenne ($RMSE$), le coefficient de détermination R^2, le taux d'échec des prédictions (FPR) et le taux d'erreurs de classification (MR).

Le deuxième chapitre est consacré à une étude détaillée sur le choix des modèles de régression, et plus particulièrement sur l'utilisation de méthodes d'ensemble pour la construction des modèles. En effet, ces méthodes sont apparues assez récemment dans le domaine général de l'apprentissage automatique et leur utilisation n'a pas encore été explorée dans le contexte spécifique d'une stratégie de test indirect. Les méthodes d'ensemble permettent de construire des modèles plus complexes, susceptibles de mieux représenter la relation entre les performances du circuit et les mesures indirectes disponibles. En outre, ces méthodes permettent de limiter la dépendance des modèles à la taille de la population d'apprentissage et à la structure des données d'apprentissage. Concrètement, ces méthodes reposent sur l'apprentissage de plusieurs modèles de régres-

sion individuels ("base learners") et la combinaison de leurs résultats afin d'améliorer la stabilité et le pouvoir de prédiction du modèle final. L'idée derrière cette procédure est qu'avec une combinaison appropriée de divers modèles individuels, il devrait être possible d'exploiter les forces et de surmonter les faiblesses des modèles individuels et d'obtenir une meilleure performance prédictive globale. Les modèles construits avec une méthode d'ensemble peuvent être assemblés de nombreuses manières, avec un apprentissage séquentiel ou parallèle des différents modèles individuels, sur la base d'un seul type de modèle (construction homogène) ou de différents types (construction hétérogène).

Dans la première partie de ce chapitre, les techniques les plus courantes de méthodes d'ensemble sont d'abord présentées, à savoir les techniques de "Bagging", "Boosting" et "Stacking". Afin d'examiner les avantages des méthodes d'ensemble par rapport aux techniques de régression classiques, d'une part, et de réaliser une étude comparative entre les différentes méthodes d'ensemble d'autre part, un protocole expérimental est développé. Ce protocole se décline en quatre phases principales. La première consiste en la partition des données en deux sous-ensembles, l'un dédié à l'apprentissage et l'autre à la validation. Afin de préserver les caractéristiques statistiques de l'ensemble de données original, une technique d'échantillonnage de type Latin-Hypercube est utilisée. La deuxième phase du protocole consiste à sélectionner les mesures indirectes les plus pertinentes en utilisant le sous-ensemble d'apprentissage. La sélection repose sur une technique de recherche itérative appelée "Sequential-Forward-Selection" (SFS). Le principe consiste à construire itérativement des modèles de régression en sélectionnant une mesure indirecte à chaque itération. La mesure indirecte sélectionnée à chaque itération est celle qui, parmi toutes les combinaisons possibles, génère le modèle avec l'erreur de prédiction minimale (score *RMSE* le plus bas). Dans ce travail, nous avons mis en œuvre une telle procédure en utilisant l'algorithme MARS pour construire les modèles de régression et en limitant la sélection à un maximum de 15 mesures indirectes. La troisième phase du protocole concerne la construction des modèles, en se basant sur les mesures indirectes sélectionnées dans la phase précédente. Dans ce travail, nous avons considéré trois type de modèles de régression classiques (MLR, MARS, SVM) et cinq méthodes d'ensemble (deux basées sur une technique de Stacking, deux sur une tecnhique de Boosting et une basée sur la technique de Bagging). Finalement, dans la dernière phase du protocole concerne l'évaluation des différents modèles. Pour cela, la prédiction des circuits contenus dans l'ensemble de validation est réalisée pour chacun des modèles construits dans la phase précédente, et les mesures d'évaluation classiques sont calculées.

Ce protocole est ensuite appliqué à une étude de cas spécifique, à savoir un amplificateur RF faible bruit pour lequel nous disposons de données de test sur plus de 3850 circuits. Les résultats montrent que l'utilisation des méthodes d'ensemble permet une amélioration de la performance globale des modèle de prédiction. Les résultats montrent également que les performances des modèles de prédiction classiques peuvent être égalées tout en utilisant un ensemble réduit d'instances d'apprentissage lorsque les méthodes d'ensemble sont mises en œuvre. Enfin, concernant la comparaison des

différentes méthodes d'ensemble, la technique de Stacking apparaît comme la plus performante.

Néanmoins, cette étude soulève des questions quant à la pertinence des métriques classiques utilisées pour évaluer la qualité d'un modèle en termes de qualité d'ajustement (R^2), précision ($RMSE$) et fiabilité (FPR). En effet, les résultats montrent que ces différentes métriques sont indépendantes les unes des autres, mais il est également difficile de les relier au taux d'erreurs de classification (MR). Cette dernière métrique apparaît par ailleurs pessimiste par rapport à l'efficacité réelle qui sera obtenue lors du test de production. Une nouvelle métrique (T-MR pour "Trusted Misclassification Rate") a été introduite, qui permet de mieux évaluer la capacité d'un modèle de prédiction à effectuer une classification correcte en tenant compte de l'incertitude de mesure RF classique.

Le troisième chapitre est consacré au développement d'une solution d'implantation originale basée sur un flot de test adaptatif. En effet, l'un des principaux problèmes qui limitent aujourd'hui le large déploiement de la stratégie de test indirect dans l'industrie est que les algorithmes d'apprentissage automatique utilisés pour construire les modèles de régression sont perçus comme une boîte noire et induisent souvent un manque de confiance dans les résultats de prédiction. Pour faire face à ce problème, une stratégie de test adaptatif peut être adoptée. En effet traditionnellement, le contenu des tests, le flot de test et les limites de test sont fixés de manière statique, ce qui signifie que toutes les pièces sont testées de la même manière, indépendamment de leurs performances individuelles. Dans le cadre d'un test adaptatif, le contenu, le flot ou les limites des tests peuvent être modifiés pour chaque pièce en fonction de données intermédiaires issues du test. L'idée est de mettre en œuvre une telle stratégie en définissant un flot de test à deux branches, la première branche correspondant au test indirect et la deuxième au test RF conventionnel. Lors du test de production, une première évaluation du circuit est réalisée par la première branche en utilisant les mesures indirectes. Cette évaluation est accompagnée d'une information sur la confiance accordée au résultat. Si la confiance est suffisamment élevée, les prédictions sont considérées comme fiables et le circuit est classé comme bon ou mauvais uniquement sur la base des prédictions réalisées par le test indirect. Si la confiance est insuffisante, le circuit est alors dirigé vers la deuxième branche où il est soumis à un test de spécification standard, c'est-à-dire que les mesures RF conventionnelles sont effectuées et le circuit est classé comme bon ou mauvais sur la base de ces mesures. L'hypothèse sous-jacente à cette approche est que la grande majorité des circuits seront triés par la première branche, et que seule une petite fraction des circuits doit passer par la deuxième branche. Cette approche offre ainsi une plus grande confiance dans la qualité et l'efficacité du test, tout en maintenant une réduction significative des coûts de test.

Dans ce chapitre, un état de l'art des solutions proposées dans la littérature selon cette approche est tout d'abord réalisé. Une nouvelle solution d'implantation est alors proposée, basée sur la définition d'une zone de tolérance autour des limites du test. En effet, les expériences réalisées dans le chapitre précédent ont montré que presque

tous les circuits mal classés sont des circuits dont la valeur prédite est proche d'une limite de test, alors que des décisions correctes sont prises pour les circuits dont la valeur prédite est éloignée des limites de test. Par conséquent, la proposition consiste à établir la confiance en examinant l'emplacement de la valeur prédite par rapport à la zone de tolérance définie autour d'une limite de test. Les principaux atouts de cette solution sont sa simplicité et sa capacité d'adaptation à différentes contraintes industrielles. Pour cela, la taille de la zone de tolérance établie autour de la limite d'essai est un paramètre crucial. En faisant varier cette taille, il est possible d'explorer différents compromis entre le coût et la qualité du test.

Dans la deuxième partie du chapitre, le protocole expérimental développé dans le premier chapitre est étendu afin d'inclure cette nouvelle option. Le nouveau protocole comporte notamment une phase supplémentaire spécifique à l'implémentation d'un flot de test adaptif, qui réalise une exploration de l'influence de la taille de la zone de tolérance sur la qualité du test (exprimée par le taux d'erreurs de classification) et le coût du test (exprimé par le pourcentage de circuits devant être soumis à un test RF conventionnel). Un autre raffinement du protocole est également introduit, qui concerne l'utilisation optionnelle d'un filtre permettant d'exclure de l'ensemble d'apprentissage les circuits présentant des caractéristiques éloignées de la distribution statistique de la population générale, ce qui est généralement recommandé dans la littérature.

Dans la dernière partie du chapitre, le protocole expérimental est appliqué à un circuit RF pour lequel nous disposons d'un large volume de données de test (plus de 26700 circuits testés). Une analyse de l'utilisation du filtre optionnel sur la composition de l'ensemble d'apprentissage est tout d'abord réalisée, pour différentes sévérités du filtre. L'efficacité du flot de test adaptatif à deux branches est ensuite évaluée et comparée à l'implémentation classique de test indirect. Les résultats montrent qu'il n'est pas pertinent de réaliser un filtrage de la population d'apprentissage car les modèles construits sur des populations filtrées se révèlent moins robustes et moins précis une fois évalués sur l'ensemble de validation, par rapport aux modèles construits sur la population initiale. Les résultats démontrent aussi clairement le bénéfice apporté par la stratégie de test adaptative. En effet, une très bonne qualité de test peut être atteinte tout en préservant une réduction substantielle des coûts de test. Plus précisément pour le cas d'étude considéré, le taux d'erreurs de classification atteint par une mise en œuvre classique de la stratégie de test indirect reste supérieur à quelques pourcents, dans les meilleures conditions. En utilisant le flot de test adaptatif, un taux d'erreurs de classification inférieur à quelques dixièmes de pourcents peut être atteint avec moins de 25% des dispositifs qui doivent passer par un test de spécification standard.

Finalement, le dernier chapitre de la thèse ouvre une nouvelle perspective d'exploitation de la stratégie de test indirect, qui concerne le contrôle des performances d'un circuit dans son application. En effet, en raison des effets du vieillissement, et en particulier dans les nouveaux nœuds technologiques, la nécessité de surveiller en ligne les performances d'un dispositif n'a jamais été aussi importante. Tout au long des travaux précédents, la stratégie de test indirect a été envisagée comme une alternative aux

mesures de performances RF classiquement réalisées pour le test en volume des circuits lors de leur sortie de production. Dans ce chapitre, nous réalisons une étude prospective sur les potentialités de la stratégie de test indirect pour un contrôle en ligne des performances du circuit au cours de sa durée de vie.

La première partie du chapitre introduit le principe d'un contrôle en ligne des performances basé sur la stratégie de test indirect et présente les adaptations nécessaires. Comme dans la stratégie classique, le principe consiste à établir, durant une phase d'apprentissage, un modèle de régression liant des mesures indirectes à une performance du circuit. La principale différence est que les mesures indirectes disponibles pour la construction des modèles doivent être significatives non seulement des effets des variations du procédé de fabrication, mais également des phénomènes de dégradation ou de défaillance susceptibles de se manifester dans le temps. Les modèles établis seront ensuite utilisés au cours de la vie du circuit pour réaliser une prédiction embarquée et vérifier qu'il n'y a pas de dégradation par rapport aux performances initiales. Il est pour cela nécessaire que le circuit ou le système soit équipé des ressources nécessaires à la réalisation des mesures et au calcul des prédictions. En particulier, le circuit doit disposer d'une infrastructure permettant l'accès aux différents nœuds ou structures internes impliqués dans les mesures indirectes utilisées dans les modèles. Le système doit par ailleurs disposer de ressources de numérisation pour convertir les valeurs analogiques mesurées dans le domaine numérique. Finalement, le système doit être muni (i) de ressources mémoire afin de stocker les coefficients des modèles établis pendant la phase d'apprentissage, les performances initiales du circuit ainsi que les valeurs de seuil indiquant une dégradation, et (ii) de ressources de calcul afin de réaliser la prédiction de la performance et vérifier que la différence entre la performance initiale et la performance estimée en fonctionnement est inférieure aux seuils définis. Dans ce travail, nous privilégions l'utilisation d'un modèle MLR en raison de sa simplicité et de sa rapidité d'exécution avec des ressources mémoire et de calcul standards. Nous proposons toutefois une solution originale pour améliorer la qualité de ce type de modèle. L'idée consiste à enrichir l'espace des mesures indirectes susceptibles d'être utilisées pour la construction du modèle, d'une part en appliquant des transformations non-linéaires simples sur les mesures indirectes, et d'autre part en considérant des interactions entre deux mesures indirectes.

Afin de développer une preuve de concept, un cas d'étude qui possède toutes les caractéristiques requises est présenté dans le deuxième chapitre. Il s'agit d'un émetteur-récepteur RF pour lequel nous souhaitons réaliser un contrôle en ligne du niveau de puissance délivré lors de l'émission du signal RF. Cet émetteur-récepteur dispose d'une infrastructure de test permettant d'accéder à onze mesures indirectes, d'un convertisseur analogique-numérique 12 bits, d'un processeur intégré (Cortex-M4 ARM), ainsi que d'un grand nombre de registres internes de configuration pouvant être utilisés pour la calibration. Cependant, ces ressources n'ont pas été pensées pour être spécifiquement utilisées dans le contexte d'une stratégie de test indirect. Une campagne de mesures a été réalisée afin de déterminer s'il est possible d'utiliser les registres internes de configuration pour émuler une dégradation du niveau de puissance émise et si les mesures

indirectes sont influencées par ces différentes configurations. Une base de données de test correspondant à 1024 observations a été constituée. Les résultats montrent que la programmation des registres internes de configuration permet d'engendrer une variation de la puissance émise, mais que seules deux mesures indirectes parmi les onze sont affectées. La situation est donc loin d'être parfaite, mais elle est considérée comme suffisante pour réaliser une étude de preuve de concept.

Dans la troisième partie du chapitre, l'élaboration du modèle pour prédire une variation du niveau de puissance émise est détaillée. L'intérêt d'utiliser un modèle MLR enrichi à l'aide de transformations non-linéaires appliquées au préalable sur les mesures indirectes est tout d'abord établi. La mise en œuvre pratique sur le cas d'étude considéré est alors présentée. La solution retenue implique quatre mesures indirectes ; le modèle construit présente une erreur quadratique moyenne *RMSE* proche de 0,4 dB. Sur la base de ces résultats, la gamme de détection est établie, en distinguant la gamme de détection possible (à partir d'une dégradation supérieure à 1,25 dB) et la gamme de détection certaine (pour une dégradation au-delà de 2,5 dB).

La dernière partie du chapitre est consacrée à la mise en œuvre pratique de la prédiction embarquée sur le cas d'étude considéré. Les premiers résultats révèlent une dispersion importante sur les valeurs prédites pour le niveau de puissance émise, ne permettant pas une détection certaine de la dégradation de performance. Etayée par une étude théorique, cette dispersion s'explique par le fait que la résolution complète du convertisseur analogique-numérique présent dans le circuit ne peut pas être exploitée, compte tenu des limitations de sa dynamique de mesure. Pour pallier cette difficulté, une technique de moyennage est mise en place pour augmenter la résolution effective du convertisseur. Avec ce moyennage, les résultats de prédiction embarquée montrent qu'il est possible de détecter une détérioration de la performance du circuit, validant donc la preuve de concept.

Au cours de cette thèse, nous avons examiné de nombreux aspects liés à l'implantation d'une stratégie de test indirect pour les circuits intégrés analogiques et RF. Des options intéressantes ont été développées afin d'améliorer la confiance dans cette stratégie ainsi que son efficacité, qu'il s'agisse de l'utilisation de méthodes d'ensemble pour construire les modèles de régression ou encore de la mise en œuvre d'un flot de test adaptatif. Nous avons également exploré un aspect totalement novateur, à savoir la possibilité d'effectuer un suivi des performances en ligne basé sur une stratégie adaptée du test indirect. Tous les résultats présentés dans cette thèse ont été évalués en utilisant des données de tests industrielles sur différents circuits RF, étayant pleinement les innovations développées.

Introduction

Checking whether an IC meets its specifications after the manufacturing process is an essential task to guarantee the device quality. However, this test process has a strong impact on the total cost of the product. This is particularly true for Analog and RF circuits that require complex and expensive test equipment with a long testing time to evaluate the circuit specifications. An interesting approach to reduce the testing costs is to adopt an indirect test strategy. The idea is to measure parameters that require only low-cost test resources and to correlate these measurements, called Indirect Measurements (IMs), with the device specifications. These correlations are often established using machine-learning algorithms.

The general purpose of this PhD is to establish a methodology for an efficient implementation of the indirect test strategy for Analog/RF Integrated Circuits. The objective is to assist and guide the test engineer in its practical choices for an efficient implementation and ensure a high level of confidence in the implemented test flow. The PhD report is divided into four chapters.

The first chapter is an overview of the indirect test strategy. We introduce the concept of an indirect test strategy. Then, we present the different classical prediction models used in previous studies. We also introduce the main techniques for indirect measurement selection. In addition, we present the various performance evaluation metrics that are used in the context of an indirect test strategy. Finally, we discuss the existing limitations in the implementation of such a strategy.

In the second chapter, we introduce the concept of the Ensemble Learning. We present the different techniques and characteristics of the existing Ensemble Learning methods. Moreover, we introduce our experimental protocol to evaluate the benefits of using Ensemble Learning methods over the classical prediction models and to complete a comparative study between the different existing Ensemble Learning methods. Then, we present and analyze the results of our study and we introduce a new evaluation metric called Trusted Misclassification Rate (T-MR).

In the third chapter, we examine a novel two-tier adaptive test flow. We begin by introducing the concept of an adaptive test flow in the context of an indirect test strat-

egy. Then, we present our proposed solution. In addition, we develop an experimental protocol for a case study. Moreover, we present the measurement campaign and the data preparation steps. Finally, we study the efficiency of the two-tier adaptive test flow and present the results of our experiment.

In the fourth chapter, we propose an adapted strategy, based on the indirect test one, to perform an on-line monitoring of the device performance within its application. Then, we present the necessary criteria for such an adaptation. Furthermore, we introduce the case study used as a proof-of-concept. In addition, we discuss the development of the model and its implementation. Finally, we present a theoretical study on the impact of the ADC resolution on the prediction accuracy as well as the results achieved while performing an embedded prediction.

Finally, the main contributions of this thesis are summarized in the conclusion and perspectives for future work. This thesis is a collaboration between LIRMM (Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier) and NXP Semiconductors in the framework of the European project HADES ("Hierarchy-Aware and secure embedded test infrastructure for Dependability and performance Enhancement of integrated Systems").

Chapter 1

Indirect Test for Analog/RF IC

1.1 Introduction

Testing Integrated Circuits (ICs) is a crucial step in the production process as it ensures the quality of manufactured devices and verifies their reliability during their life-cycle. 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 less evolved 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 [1]. Furthermore, recent design trends tend to experiment with heterogeneous systems and the notion of System-in-Package and 3D devices. Such advances raise new technical difficulties for the testing process in terms of access to the internal components in order to provide stimuli for specific inputs and the ability to read the device response on its outputs, which will of course result in additional test costs. Finally, when considering RF ICs, it would be complicated to rely entirely on wafer-level specification-based testing for RF signals due to probing complexity [2] and limited resources [3].

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. Built-in-Self-Test or DFT (Design For Testability) solutions often reduce the required external resources, 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.

This chapter is organized as follows. Section 1.2 presents the concept of an indirect test strategy. Section 1.3 describes the synopsis of this test strategy. In Section 1.4, we present the commonly used classical prediction algorithms in order to choose the appropriate machine-learning algorithm for our study. Section 1.5 explains the various available techniques to choose the most pertinent and important indirect measurements. In Section 1.6 we define the metrics that we will use to evaluate our indirect test strategy. In Section 1.7 we present the initial limitations faced with the indirect test strategy. Finally, Section 1.8 concludes the chapter.

1.2 Concept of an Indirect Test Strategy

Indirect test for analog/RF integrated circuits was firstly introduced in [4] as alternate test. The main motivation behind such an approach is to alleviate the burden and relax the constraints on the industrial test equipment to process conventional performance specification measurements, which require additional dedicated and expensive equipment. The underlying concept of an indirect test strategy is that the process variations exhibited in the fabrication process that affect the device performances will also affect non-conventional low-cost indirect parameters easily measured by low-cost test equipment. Thus, it is possible to find and establish a correlation between the indirect parameter space and the performance specification space. As a result, the intention of establishing a correlation is to test only indirect parameters to verify the performance of the device under test.

However, the relation between these two sets of measurements is usually complex and not always easy to identify through analytical functions. One solution to overcome this problem is to utilize the computing powers of machine learning algorithms. The implementation of a machine learning algorithm can be under two distinctive forms in the context of an indirect test strategy; it could be either considered as a classification or a regression problem.

1.2.1 Classification-oriented Indirect Test

In the case of a classification-oriented indirect test, the idea is to establish a decision boundary that separates good from faulty circuits. This decision boundary is determined within the indirect measurement space. This approach has been previously presented in [5–7].

Of course, such an approach is only possible when the test limits are available to be able to establish a model that differentiates between these two classes. Indeed, implementing a classification-oriented indirect test is deemed as a fast strategy to classify

new devices as either faulty or good circuits while only using rapidly executed low-cost indirect measurements. However, once the new device is classified, this approach does not offer any capabilities to diagnose the results of such a classification. Moreover, usually due to the life cycle of new devices, the specification test limits may change, which implies to re-define and re-establish a new decision boundary in the indirect measurement space to classify circuits as good or faulty.

1.2.2 Prediction-oriented Indirect Test

The prediction-oriented indirect test is another implementation of an indirect test strategy, which has been previously explored in [8–10]. Instead of establishing a decision boundary, like in the case of a classification-oriented indirect test, in this approach the target is to establish a regression function that can predict the value of the device performance by building a regression function that maps the values of indirect measurements (IM_i) into the performance measurements (P_j) as expressed in Equation 1.1, thus dispensing the need of retaining the performance specification limits while establishing the indirect test strategy.

$$f_{IM \to P} : [IM_1, ..., IM_l] \to [P_1, ..., P_N]. \tag{1.1}$$

In fact, this strategy has several advantages over the previous strategy. The main advantage is that there is no need to re-establish the regression function that maps the indirect measurement space to the performance specification space each time the specification test limits vary during the life cycle of the device. Moreover, due to the produced performance estimation through the established regression function, it is plausible to diagnose, and interpret the indirect test strategy efficiency. Thus, in our work we have adopted the prediction-oriented test approach due to its advantages over classification-oriented strategy, and we present the prediction-oriented indirect test synopsis in the following section.

1.3 Synopsis of an Indirect Test Strategy

When implementing an indirect test strategy based on machine learning algorithms, the aim is to replace the conventional specification based testing approach. Hence, the machine learning model should satisfy different important criteria. The regression model in which we estimate the performance of the device should be of high accuracy, and represent correctly the relationship between the device's specifications and the indirect measurements. Furthermore, the regression model has to predict the performance specification of the device under test (DUT) during the production testing phase in a reliable manner. Finally and most importantly, the regression model should utilize the minimum number of indirect measurements as possible to efficiently reduce the test cost.

The indirect test synopsis can be divided into two distinct phases, namely learning and production testing phases, as illustrated in Figure 1.1. The idea is to establish during the learning 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 indirect measurements are performed on a set of learning devices. It is imperative that the set of learning devices is representative of the set of devices tested at the production phase. Thus, a machine-learning algorithm is trained on the set of learning devices to build a regression model that maps the indirect parameter space to the performance parameter space. Then, 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 learning phase.

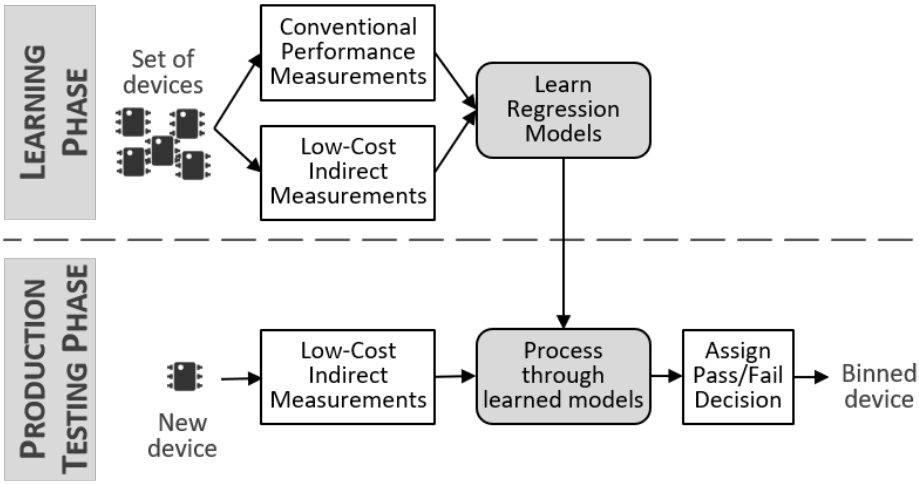


Figure 1.1: Indirect test synopsis [11]

1.4 Classical Regression Algorithms

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), Support Vector Machine (SVM), and Decision Trees (DT). The fundamentals of these models are briefly described hereafter.

1.4.1 Multiple Linear Regression

A 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) [12]. Once trained, the model

predicts new instances by simply computing a weighted sum of the input variables expressed in Equation 1.2, where x_i represents the different input variables and θ_i are the different model parameters, which include the bias term of the regression function θ_0. The parameters of the model are computed in a way that minimizes the RMSE (Root Mean Square Error, see Section 1.6) score on the training dataset.

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + ... + \theta_n x_n \tag{1.2}$$

The main interest of this model is that it gives a clear idea of how the inputs affect the output. Moreover, thanks to its extreme simplicity, it is very fast to compute. However, because it assumes only linear relationship between the input and output variables and is considered as a parametric regression function, it might not be appropriate to correctly represent complex data, since it tends to under-fit the data. An example of a simple linear regression model is presented in Figure 1.2.

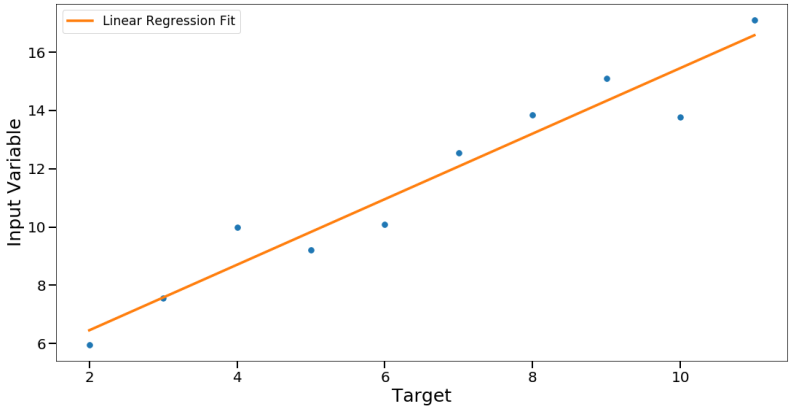


Figure 1.2: Simple Linear Regression

1.4.2 Multivariate Adaptive Regression Splines

A more refined model is a MARS model [13], which is based on a non-parametric regression. It can be considered as an extension of linear models. Nonetheless, it includes automatic modeling of non-linearities and interactions between variables. In particular, the technique involves the partitioning of the input space into several regions, each one with its own linear regression equation. The algorithm automatically computes the different parameters related to the partitioning of the input space and the combination of the variables and what is called hinge functions expressed with Equation 1.3. An example is illustrated in Figure 1.3.

$$h(x-t) = [x-t]_+ = \begin{cases} x-t, & x > t \\ 0, & else \end{cases} \tag{1.3}$$

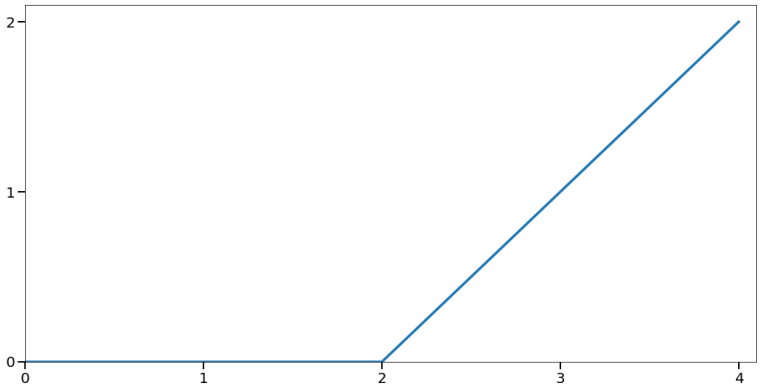


Figure 1.3: Hinge Function

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 MLR models.

1.4.3 Support Vector Machines

SVM is a supervised machine learning algorithm that can be used for both classification or regression challenges. The general objective of the SVM algorithm is to find a linear hyper-plane which separates the data into classes for a classification problem, and to fit the data within this hyper-plane for a regression problem. However, if a linear hyper-plane cannot be found to fit or separate the data, the algorithm can exploit the built-in kernel methods that transform the data into a higher order dimensional space by creating new features allowing the algorithm to find a linear hyper-plane and solve non-linear problems. Polynomial and Gaussian Radial Basis Function (RBF) are the mostly used built-in kernels, and are defined in [12].

The characteristics of the linear hyper-plane, its dimensions and its support vectors are only determined based on the training data. Thus, the number of coefficients that defines the hyper-plane will increase with the number of training instances and this might entail an additional computational complexity. In Figure 1.4, an example of a Linear SVM regression model is presented by its hyper-plane and the different support vectors. More details can be found in [14].

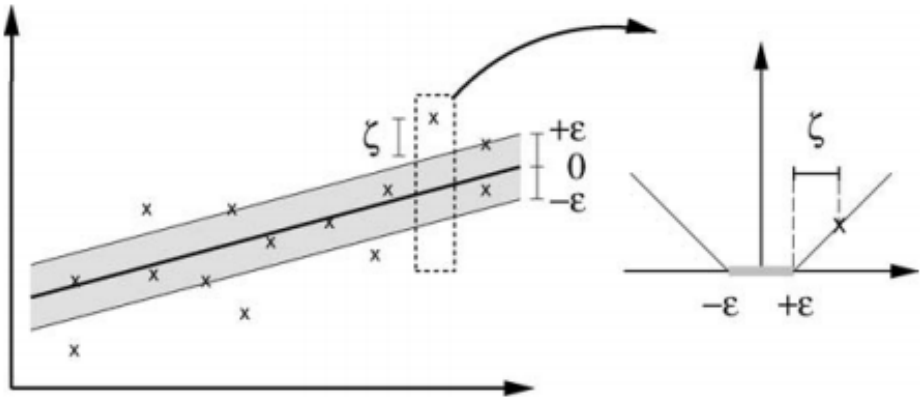


Figure 1.4: Linear Support Vector Regression [14]

Finally in the case of a regression problem, the prediction of new instances will depend on the support vectors and the coefficients of the model. The value of a new instance can be calculated through Equation 1.4, where $k(x_i, x)$ represents the kernel transformation built in the algorithm, the coefficients along with the support vectors are defined within $\alpha_i^* - \alpha_i$, and b represents the bias term in the regression function.

$$f(x) = \sum_{i=0}^{l} (\alpha_i^* - \alpha_i) k(x_i, x) + b \tag{1.4}$$

1.4.4 Decision Trees

Similar to SVM algorithms, Decision Trees are considered as a flexible machine learning algorithms, where both regression and classification problems could be solved. This algorithm identifies ways to split the dataset across the available features. Moreover, the decision rules are generally built on the bases of if-then-else statements, as the example presented in Figure 1.5.

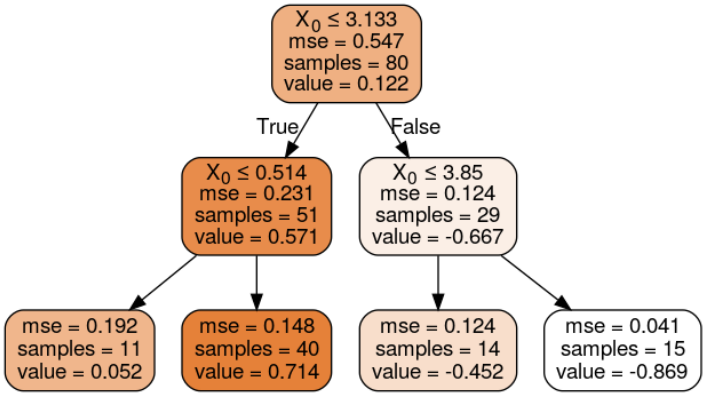


Figure 1.5: Decision Tree Decision Rules

Unlike MLR, Decision Trees are considered as non-parametric supervised machine learning algorithms since the number of parameters is not determined prior to training the model. In such circumstances, the model is free to replicate the training data, which might lead to over-fitting. However, such circumstances could be avoided by controlling the model's hyper-parameters which could introduce a regularization factor and control its degree of freedom. Number of hyper-parameters can control the model. One of them is the depth of the decision tree: if left uncontrolled, the tree and its decision rules will continuously expand until there are no samples left in the dataset. Furthermore, the minimum number of samples required to split the data and create a decision rule can also be specified. Indeed, tuning this types of models requires deep understanding of the effect of each hyper-parameter on its fit. Further details can be found in [12].

Finally, the effect of one of the Decision Tree parameters (max_depth) is presented in Figure 1.6 as an example. It shows how the fit of the data could change as we vary one of the model's hyper-parameters.

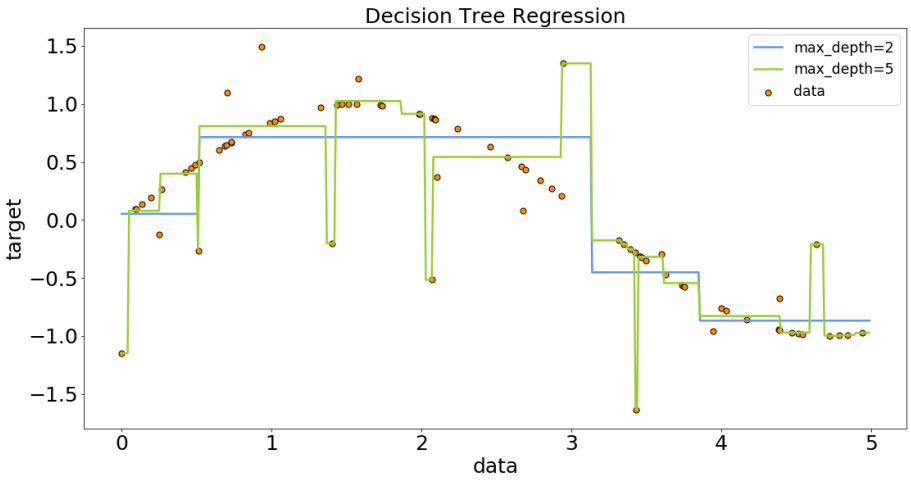


Figure 1.6: Max Depth parameter effect

1.5 Indirect Measurement Selection

In the previous section, we briefly described the most common regression algorithms used in the context of indirect test strategy. Nonetheless, regardless of the model type or complexity, the quality of the prediction model will be hugely dependent on the available indirect measurements in the dataset. Thus, the choice of the indirect measurements is going to affect greatly the performance achieved through an indirect test strategy, which raises the importance of finding pertinent indirect measurements. Nevertheless, the pertinent indirect measurements can vary from one product to another,

or even from one specification to another for the same product.

Furthermore, we should ensure that the used indirect measurements are information-rich and correlate with the circuit specifications. Primarily, the test engineer will depend on the designer expertise and knowledge of the circuit under test to use the ad-hoc indirect measurements. The authors in [15] have used standard final DC tests that are usually performed for each manufactured device, which can thus be easily added to the list of indirect measurements without additional costs. Moreover, it is also possible to use internal DC probes along with a DC test bus to measure some pre-defined internal nodes, as presented in [16]. Furthermore, designers could also include various types of built-in sensors that could be deemed as additional indirect measurement, as proved in [17, 18]. Finally, the authors in [9] have explored the possibility of changing the test conditions for the various types of indirect measurements, in order to increase the total number of indirect measurements.

Once all the possible indirect measurement candidates have been generated for a specific product, it becomes feasible to construct a prediction model that correlates the circuit specifications with the rich collection of indirect measurements. However, the effectiveness of this process comes into question for various purposes. Firstly, we have to remember the main motivation behind the indirect test strategy, the aim is to reduce the test cost for analog and RF integrated circuits. Thus, having a substantial collection of indirect measurements would keep the test cost at a high level, which would nullify the expected gain of an indirect test strategy. In addition, building a complex prediction model comprised of numerous input variables will increase the computational complexity of the task. Furthermore, some of the generated indirect measurements might not contain any additional information or even duplicate or correlate well with other indirect measurements, which will reduce their importance. Finally and most importantly, while increasing the number of indirect measurements will diversify the available collection that could be used to predict the circuit specifications, it will force us to face the curse of dimensionality, which is a well documented phenomenon in the domain of machine learning [19]. This problem will affect the performance of the regression model where the generalization error will eventually start to increase as the dimension of the input variables increases, tending to over-fit the training data.

Consequently, limiting the number of indirect measurements is beneficial on several fronts, and it raises the importance of selecting the most pertinent indirect measurements among the collection of the 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. For this, various algorithms have been proposed, which can be divided into three categories, namely filters, wrappers and embedded methods [20], as well as any hybrid algorithm that combines the above methods.

1.5.1 Filter Methods

Filter methods select pertinent features based on their statistical characteristics. These methods are thus independent of the machine learning model and are considered as a pre-processing step, since the selection process begins before training the model and since it does not take into consideration the model's performance.

Mainly, the motivation behind applying these types of methods is justified since they are easy to implement, simple to comprehend, and most importantly computationally inexpensive. Basically, filter methods are very good tools to eliminate redundant, irrelevant or duplicated features. Filter methods could range from the very basic filters that eliminate constant and quasi-constant features, or features that are highly correlated with each other, to statistical and correlation-based filters that could be univariate or multivariate.

Feature selection based on Pearson correlation

The Pearson correlation describes the strength of the linear relationship between two random variables, hence it could be calculated for the different indirect measurements (input variables) with respect to the circuit specifications (target variables). It is expressed by Equation 1.5, where $\bar{x}$ and $\bar{y}$ represent the mean value of the random variables X and Y, whereas n represents the total number of samples.

$$R_{X,Y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{1.5}$$

The Pearson correlation is built on the assumption that a linear dependency exists between the two random variables and that they are normally distributed. The value of the Pearson correlation can vary between -1.0 and 1.0:

- 1.0 means positive correlation,
- -1.0 means a negative correlation,
- 0 means no correlation between the two random variables.

Actually, it is also possible to define the Pearson correlation as the covariance of the two random variables divided by the product of their standard deviation, as expressed in Equation 1.6:

$$R_{X,Y} = \frac{cov(X,Y)}{\sigma(X)\sigma(Y)} \tag{1.6}$$

Consequently, in the context of an indirect test strategy, we compute the Pearson correlation for each indirect measurement and we rank them in a decreasing order based on their score. Once ranked, it is possible to select the most relevant indirect measurements without the use of a prediction model, by choosing the highest ranked indirect measurements.

Feature selection based on Brownian distance correlation

Unlike Pearson correlation, which assumes a linear dependency between random variables, Brownian distance correlation [21] is a multivariate correlation score that is sensitive to non-linear dependencies. It has been used in the context of an indirect test strategy in [22].

Considering $(X_i, Y_i) : i = 1, 2, ..., n$ a random instance from an independent and identically distributed random vector variable (X, Y). The squared sample distance covariance $dCov_n^2(X, Y)$ is defined as the arithmetic average of the products $A_{j,i}$ and $B_{j,i}$ presented in Equation 1.7.

$$dCov_n^2(X, Y) = \frac{1}{n^2} \sum_{j,i=1}^{n} A_{j,i} B_{j,i} \tag{1.7}$$

where the products $A_{j,i}$ and $B_{j,i}$ correspond to the doubly centered distances computed from the Euclidean distance matrices $(a_{j,i}) = (\|X_j - X_i\|)$ and $(b_{j,i}) = (\|Y_j - Y_i\|)$:

$$A_{j,i} = a_{j,i} - \bar{a}_{j.} - \bar{a}_{.i} + \bar{a}_{..}, B_{j,i} = b_{j,i} - \bar{b}_{j.} - \bar{b}_{.i} + \bar{b}_{..}$$

where

$$\bar{a}_{j.} = \frac{1}{n} \sum_{i=1}^{n} a_{j,i}\ ,\ \bar{a}_{.i} = \frac{1}{n} \sum_{j=1}^{n} a_{j,i}\ ,\ \bar{a}_{..} = \frac{1}{n^2} \sum_{j,i=1}^{n} a_{j,i}$$

and

$$\bar{b}_{j.} = \frac{1}{n} \sum_{i=1}^{n} b_{j,i}\ ,\ \bar{b}_{.i} = \frac{1}{n} \sum_{j=1}^{n} b_{j,i}\ ,\ \bar{b}_{..} = \frac{1}{n^2} \sum_{j,i=1}^{n} b_{j,i}$$

The sample distance variance $dVar_n(X)$ is a special case of distance covariance when the two variables are identical and is given by the square root of:

$$dVar_n^2(X) = dCov_n^2(X, Y) = \frac{1}{n^2} \sum_{i=1}^{n} A_{j,i}^2 \tag{1.8}$$

Finally, the distance correlation $dCor_n(X, Y)$ of two random variables(X, Y) is obtained by dividing their distance covariance by the product of their distance standard deviations (square root of distance variances):

$$dCor_n(X,Y) = \frac{dCov_n(X,Y)}{\sqrt{dVar_n(X)dVar_n(Y)}} \tag{1.9}$$

In a similar manner to the Pearson correlation, the most relevant indirect measurements are the most highly ranked measurements.

1.5.2 Wrapper Methods

The filter methods, explored previously, rely basically on the statistical characteristics of the features, and select them in an independent manner from the type of prediction model. As a consequence, the features are not selected to optimize the prediction model performance and tend to ignore feature interactions, since filter methods evaluate features individually. Thus, it would be interesting to explore feature selection methods which take into consideration the prediction performance of the model. Such methods are defined as wrapper methods.

A wrapper method is firstly based on a search strategy within the space of possible feature subsets, then each feature subset is evaluated based on the prediction model performance. Such a process entails different choices that should be specified before implementing such a method:

- Search strategy,
- Prediction model type,
- Performance evaluation metric (see Section 1.6),
- Stopping criterion.

Such methods are considered as greedy search algorithms since they tend to find the best possible feature subset that maximizes the prediction model performance, and most often will be computationally expensive in the case of an exhaustive search. On the other hand, wrapper methods can detect feature interactions and will find an optimal solution tailored for a specific type of prediction model. Finally, to reduce the computational burden of such a process, it is recommended to include a stopping criterion for the search strategy, either by monitoring the performance behavior or by choosing a desired number of features. Multiple search strategies exist, such as best-first, depth first search, hill climbing and genetic algorithms. Nonetheless, the most common strategy used in the context of an indirect test strategy is based on a best-first search, which adds the best feature candidate in each iteration. It is also known as Sequential Forward Selection (SFS) [18, 23].

The procedure of SFS starts by building a prediction model for each available feature and selecting the feature that generates the model with the best performance. At the

second iteration, a prediction model is built for each pair of features that includes the previously selected feature; 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 features reaches a maximum target limit, or the desired level of performance is reached.

1.5.3 Embedded Methods

Unlike wrapper methods in which the models are trained and then their performances are evaluated to select the feature subset, embedded methods complete the feature selection process within the training process of the prediction model. Embedded methods internally compute the features importance in terms of prediction contribution during the training phase, and in the end remove non-important features. Thus, the main advantages of such a process are the following:

- Fast computation, similar to filter methods,
- Consideration of feature interactions, similar to wrapper methods,
- Higher accuracy than filter methods,
- Less prone to over-fitting.

Tree-based models are capable of identifying the most important features while building the model since, as seen previously in the case of Decision Trees (Section 1.4.4), they tend to split the dataset based on certain features to build the decision rules. Hence, once the tree based algorithm is fully trained, it is easy to rank the importance of features based on the dataset partition decision rules. MARS model (Section 1.4.2) is another prediction model that is capable of eliminating unnecessary features while building the prediction model, as presented in [24].

Nevertheless, surely not all types of prediction models are capable of performing embedded feature selection. Another solution for linear models is the use of regularization: it is quite used in the domain of machine learning whenever we wish to penalize over-fitting, or to limit the freedom of certain parameters. Indeed, regularization also increases the model's robustness to noise and improves its generalization to prevent huge validation errors. There exists three types of regularization for linear models:

- Lasso Regression, or L1 regularization,
- Ridge Regression, or L2 regularization,
- Elastic Nets L1/L2 regularization.

For interested readers, the full explanation and the differences between those three types of regularization is presented in [25, 26]. This topic is out of the scope of this chapter.

1.5.4 Hybrid Methods

Finally, after having presented the most commonly used methods to perform feature selection in the domain of machine learning and their implementation in an indirect test strategy, it is possible to compare the pros and cons of each method in order to utilize the most suitable method for a specific application. Nonetheless, it is also possible to consider hybrid methods which are constructed using two or more of the above mentioned methods. Indeed, we can create a tailored made method that could suit our application; for example we can use filter methods alongside wrappers or even wrappers alongside embedded methods. Such propositions have been explored in the context of an indirect test strategy as presented in [27].

1.6 Evaluation Metrics

Once the most pertinent indirect measurements have been selected, along with the type of regression model which of course will depend on the data and the device under test, the evaluation of the prediction model implemented within the indirect test strategy will be considered as a crucial following step. This evaluation will serve as a reflection on the efficiency of the indirect test strategy, and as an indication of confidence whenever a test engineer considers replacing the classical specification test by an indirect test strategy. There exists several evaluation metrics in the world of machine learning, however, the metrics used in the context of a prediction-oriented indirect test strategy are discussed in this section.

1.6.1 Normalized Root Mean Square Error

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 actually 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:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{1.10}$$

where y_i is the actual performance value of the i^{th} instance, $\hat{y}_i$ is the predicted performance value of the i^{th} instance and n is the number of instances. 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 manuscript, 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:

$$NRMSE = \frac{RMSE}{\bar{y}} \tag{1.11}$$

1.6.2 Coefficient of Determination

Another common metric used in statistics to evaluate the quality of a regression 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 described in Equation 1.12. The interesting thing about the R^2 score is that it is a normalized score that ranges between 0 (no correlation) and 1 (perfect correlation), therefore permitting comparison across different variables.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y_i})^2} \tag{1.12}$$

Note that the *NRMSE* score can be computed form the R^2 score as shown in Equation 1.13 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.

$$NRMSE = \frac{\sigma_y}{\bar{y}} * \sqrt{1 - R^2} = CV_y * \sqrt{1 - R^2} \tag{1.13}$$

1.6.3 Failing Prediction Rate

Another metric has been introduced in [28], which permits to quantify the prediction reliability of a model. This metric, called Failing Prediction Rate (FPR), which is represented in Equation 1.14, expresses the percentage of circuits with a prediction error that exceeds the conventional measurement uncertainty ε_{meas}.

$$\begin{aligned} FPR = \frac{1}{n}\sum_{i=1}^{n}(|y_i - \hat{y}_i| > \varepsilon_{meas})\, with\, (|y_i - \hat{y}_i| > \varepsilon_{meas}) = 1\, if\, True \\ (|y_i - \hat{y}_i| > \varepsilon_{meas}) = 0\, otherwise \end{aligned} \tag{1.14}$$

1.6.4 Misclassification Rate

When the conventional specification test is considered for the circuit performance verification, the test quality could be obtained through two key metrics: the yield loss

and the test escape. The yield loss represents the percentage of good circuits that were discarded, out of the total number of circuits. The test escape represents the percentage of circuits deemed to be functional when in reality they are faulty circuits, out of the total number of circuits. It is important to note that both of these metrics could be only computed if the test limits are available and the notion of good and faulty circuits is established.

It would be interesting if a similar metric could be used in the context of indirect test strategy. Actually, such a metric already exists and it is called the Misclassification Rate (MR); it can be computed only if the test limits are available in both prediction or classification-oriented indirect test strategies. Basically, it expresses the ratio of misclassified circuits, when the model predicts faulty circuits as good or vice versa, with respect to the total number of circuits. Hence, the MR metric expressed in Equation 1.15 could be considered as the sum of the yield loss and the test escapes metrics which reflects the test quality in the classical specification test.

$$MR = \frac{yield\ loss + test\ escapes}{Total\ \#\ of\ circuits} \tag{1.15}$$

1.7 Limitations

After introducing the basic and necessary elements of an indirect test strategy in this chapter, we have to be realistic about the prospect of adopting such a strategy, since some crucial challenges might be faced during the production test phase. Those challenges will have to be treated in order to establish a confident and efficient indirect test strategy, since we are replacing the expensive and reliable specification-based testing with low-cost indirect measurements for a huge number of devices. Generally, industry always has doubts about this strategy and does not have a full confidence in the predicted performance values, due to the lack of clarity and certainty on various aspects. For instance, the regression model is heavily dependent on the training phase, which is based most importantly on the selected indirect measurements and the number of available devices. We have to remember also that these prediction models are built using a limited set of training devices that might not reflect the actual distribution and the behavior of the device during its full life cycle.

Previous studies have been already conducted to try and alleviate those challenges, but still the indirect test strategy is far from being adopted in an industrial context. In this manuscript, we will try to improve the efficiency of the existing solutions. Our objective is to achieve and implement a complete framework, which could help any test engineer in performing an adequate comparative analysis and study of the different indirect test strategy options in order to reach a state of full confidence in the indirect test strategy.

1.8 Conclusions

The concept of an indirect test strategy has been introduced in this chapter. This test strategy is an alternative approach to the specification-based testing for analog/RF integrated circuits in order to reduce the ever-increasing testing costs. Although this strategy has been proposed and developed over the years, it has generally not been adopted in an industrial context due to the lack of confidence. Nevertheless, this strategy heavily relies on the machine learning domain and could always be improved by newly proposed techniques or models for better consideration from the industry.

The various elements required to implement an indirect test strategy were presented, such as the type of indirect test strategy, the regression models, the indirect measurement selection and finally the evaluation metrics. Indeed, different options exist within each essential element, which allows us to perform a comparative analysis and highlight the advantages and disadvantages of each option. Thus, we offer an important insight for test engineers who wish to implement an indirect test strategy.

In the following chapters, we will present our work in which we investigate different approaches to enhance the performance of the prediction models, or to improve the overall confidence in the prediction value while offering a complete framework to establish an efficient and trustworthy indirect test strategy.

Chapter 2

Ensemble Learning

2.1 Introduction

In the previous chapter, the concept of an indirect test strategy for analog/RF circuits was presented and discussed. Moreover, the predominant prediction models that are usually used in this context were described in section 1.4. However, regardless of the type of prediction model, using one prediction model for an indirect test strategy can force certain limitations, and could be problematic in some scenarios. Indeed, the performance of an individual prediction model may vary from one product, or specification to another, and the size of the training set can also impact the performance achieved by the implemented prediction model. Furthermore, the presence of extreme values in the dataset can lead to deterioration in the performance of some types of prediction models.

Therefore, the objective of this chapter is to explore the use of multiple prediction models. The chapter is organized as follows. In section 2.2, we define the concept of ensemble learning and introduce the different methods that are usually used in the literature in the domain of machine learning. An experimental setup is then introduced in section 2.3, in which we investigate the advantages of ensemble learning over classical individual models, and compare the different methods of ensemble learning. Results are summarized and discussed in section 2.4. Finally we conclude with a perspective of extending the implementation of indirect test strategy in section 2.5.

2.2 Ensemble Learning

Unfortunately, in the field of machine learning, it is quite impossible to find an ideal model, a single prediction model that can outperform others in different scenarios or circumstances. Thus, researchers began to use several prediction models to circumvent the limitations and drawbacks of using a single prediction model. The main idea of this approach is to exploit the strengths and mitigate the weaknesses of individual models. Hence, the combination of a diverse set of individual models can potentially lead to

better stability and better predictive power.

The process of using multiple individual models (base learners) and aggregating their outcomes is called Ensemble Learning. It can be applied in various forms which have been proposed in the literature [29]. The different base learners can be homogeneous, by using a single type of base learners, or heterogeneous, by using different types of base learners. Both of which can be trained sequentially or in parallel, depending on the ensemble learning algorithm and technique.

In the context of indirect testing of analog/RF circuits, ensemble learning has been firstly introduced in [30], where the implementation of the ensemble methods has been handled by the ENTOOL Matlab toolbox [31]. However, the authors in [30] used the toolbox without conducting a full comparison between the different ensemble methods, and studying the benefits of using such methods over the use of classical individual models. In addition, model redundancy which is an aspect of ensemble learning, has been investigated in [32, 33] where it is compared with the classical approach of using one individual prediction model. In this section, we wish to explore various ensemble learning algorithms and techniques. More precisely, three of the most largely used ensemble learning methods are considered, namely Bagging, Boosting, and Stacking.

2.2.1 Bagging

A first approach to obtain a diverse set of base learners consists in manipulating the original training set in order to create different random subsets. Multiple base learners can then be trained on these random subsets and aggregated by averaging their output, which will reduce the variance achieved with an individual model for the totality of the training set.

The creation of the random subsets can be achieved by re-sampling the original training set. When re-sampling is performed with replacement, it is called Bagging which stands for bootstrap aggregating. Note that, when Bagging is applied, it is possible for a training instance to be sampled several times for the same base learner. This process is illustrated in Figure 2.1; it is a parallel ensemble method based on homogeneous base learners.

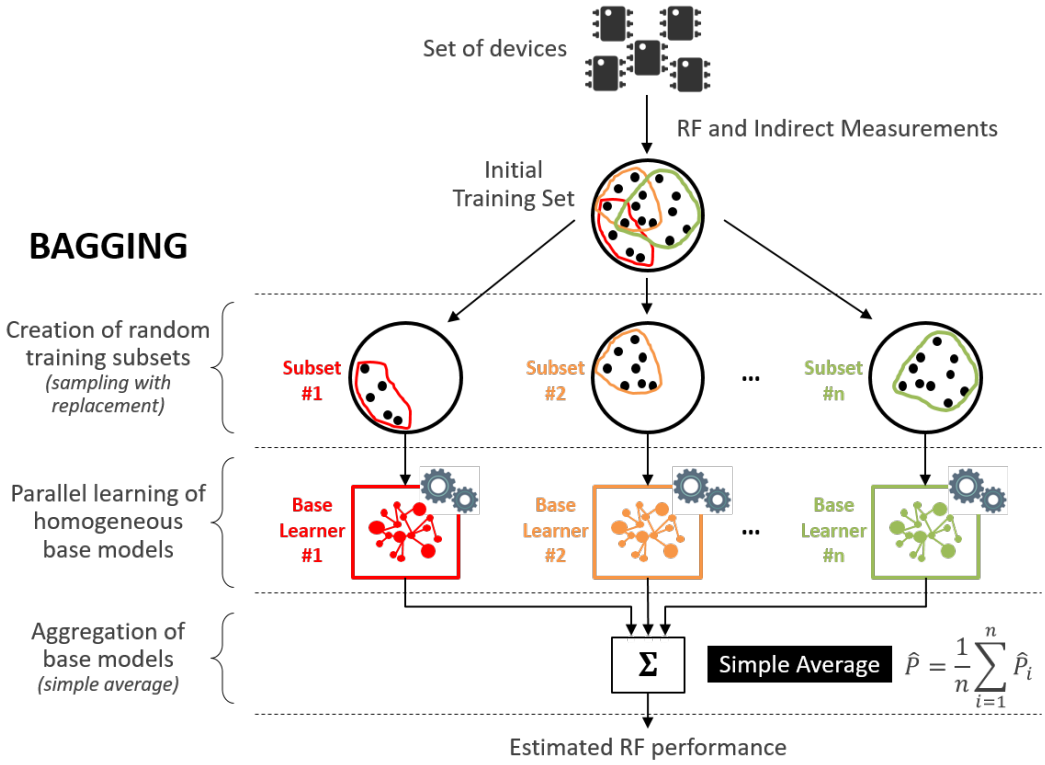


Figure 2.1: Principle of ensemble model construction with Bagging

One of the most popular Bagging algorithm is called Random Forest. It is composed of multiple decision trees (Section 1.4.4), which are trained on various sub-samples of the initial dataset and randomized by selecting a random subset of features averaged to produce the final ensemble model. However, Bagging can be applied with any type of model.

2.2.2 Boosting

Boosting is also an ensemble method based on homogeneous base learners built using a manipulation of the original training set. However contrary to Bagging, it is a sequential method where the different training sets are calculated depending on the performance of the base learner constructed in the previous iteration. The main idea of Boosting is that, at each iteration, a new model is built with the objective to correct the prediction errors of its predecessor, thus leading to a better ensemble model by focusing on the under-fitted samples at each step. Two of the most used approaches implemented in Boosting are AdaBoost and Gradient Boosting.

AdaBoost

Adaptive Boosting (AdaBoost) is implemented by learning a first predictor on the entire training set by assigning an equal weight to all the samples. In a second step, depending on the performance of the model, the weights are then updated to highlight all the under-fitted samples. Subsequently, the next-in-line model is trained on the updated training samples with their newly assigned weights. This process continues until all the prediction models have been trained. Finally, a weighted average of the different predictions is used to compute the final prediction value, as illustrated in Figure 2.2.

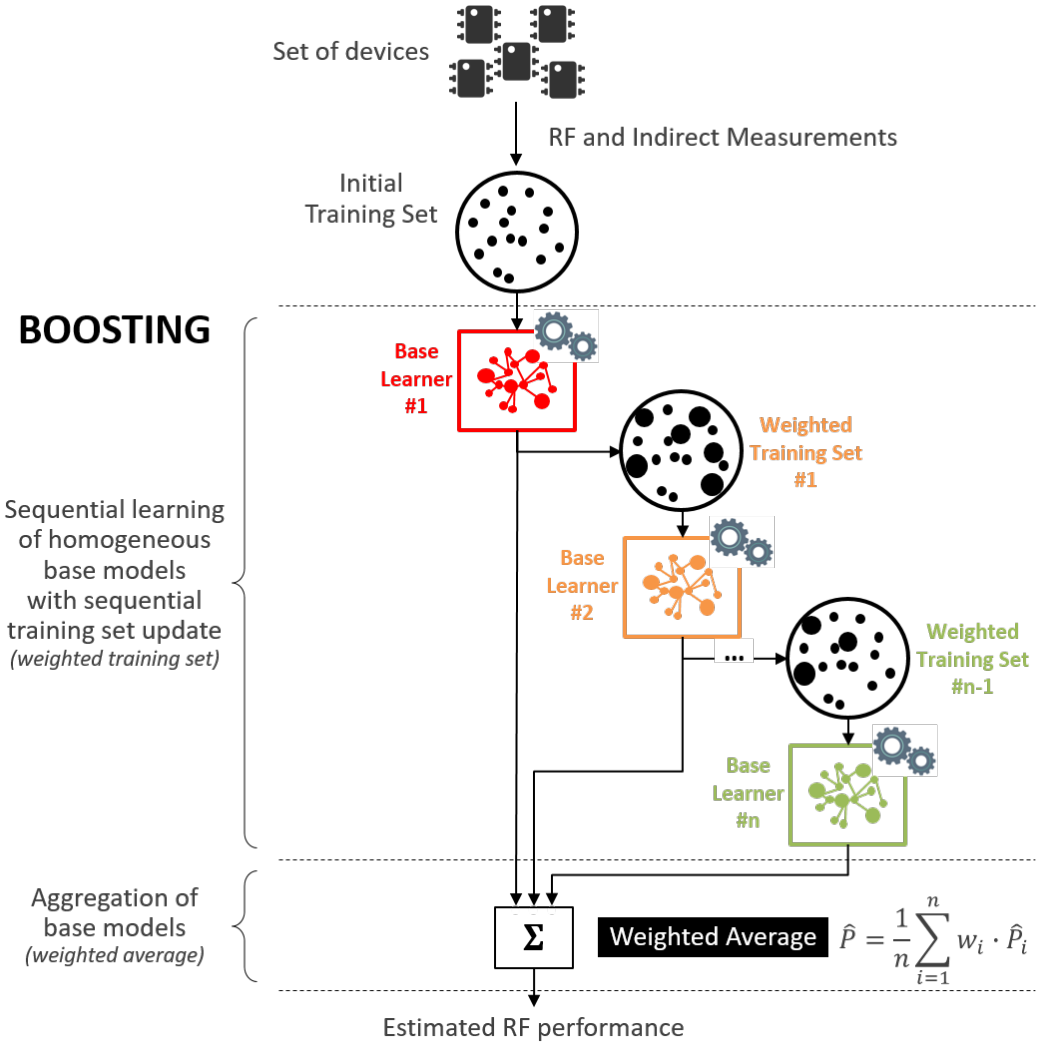


Figure 2.2: Principle of ensemble model construction with AdaBoost

Gradient Boosting

Like AdaBoost, the Gradient Boosting generates sequentially a new model that corrects its predecessor. However, instead of updating the weights of the training samples, Gradient Boosting algorithm tries to fit the residual errors made in the previous step. In this case, the final prediction value is produced by adding all the predictions obtained in every step (Figure 2.3). Although both Boosting techniques can be used with any type of predictive model, they are generally applied using decision trees as base learners.

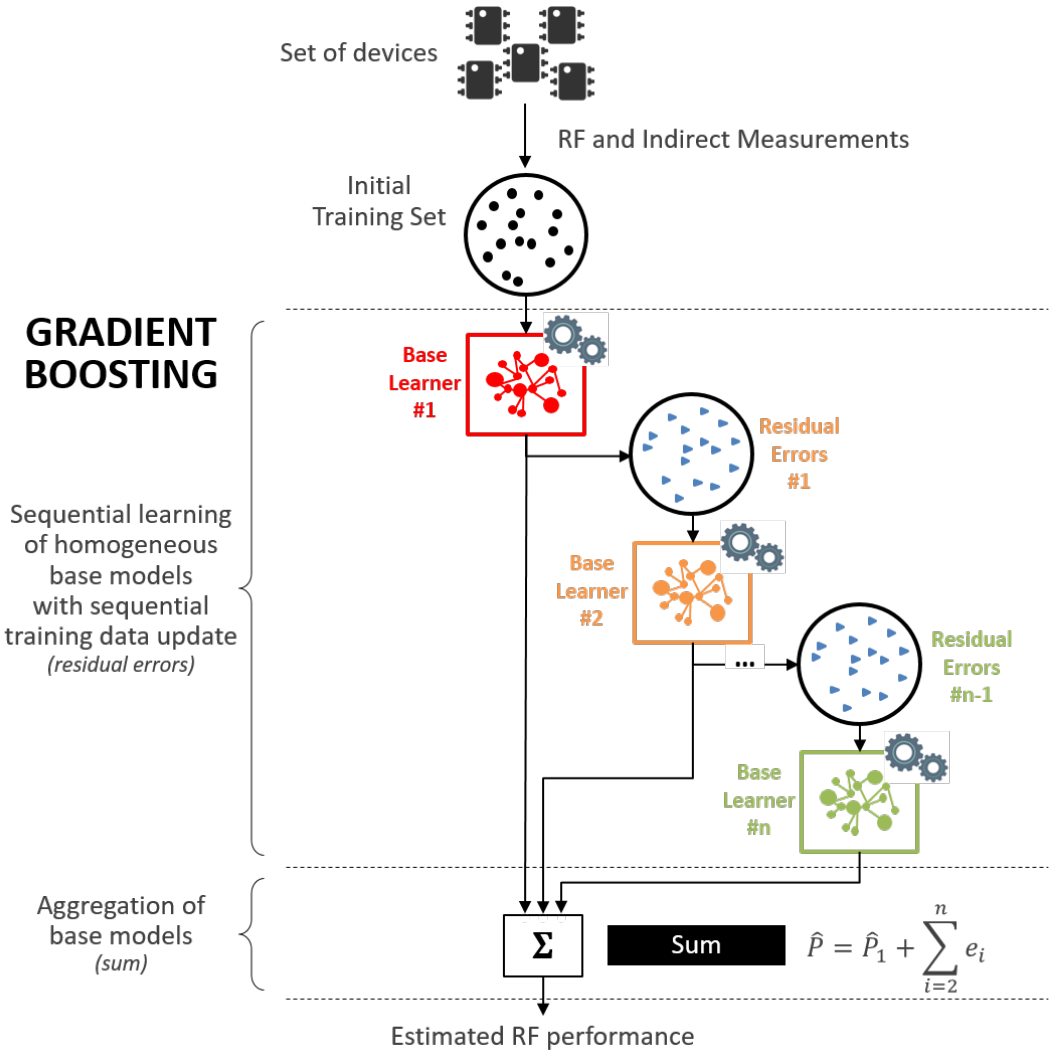


Figure 2.3: Principle of ensemble model construction with Gradient Boosting

2.2.3 Stacking

An ensemble model can also be built by using different types of base learners. Unlike Bagging and Boosting methods, Stacking (short for stacked generalization) uses heterogeneous types of base learners. It is established on a simple concept. Instead of aggregating the different predictions using a trivial function (a sum or a weighted average), a prediction model, called a meta-learner, is used to perform the aggregation of the various base learners. The diversity of base learners is achieved by varying the types of models. The meta-learner is trained by using the predicted values of the base learners as input features, as illustrated in Figure 2.4. 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.

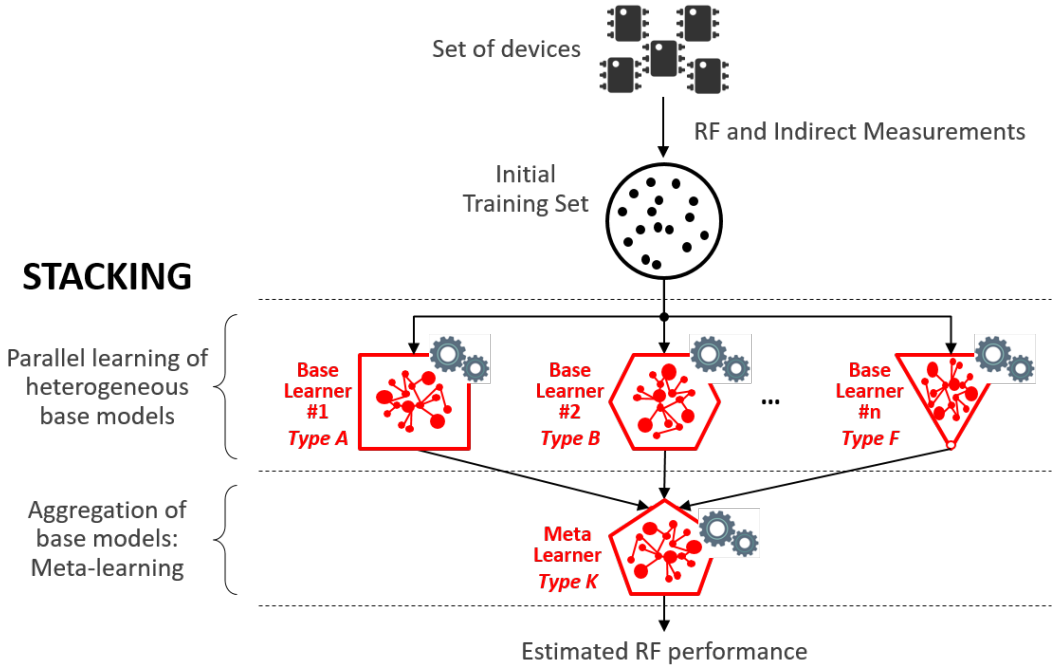


Figure 2.4: Principle of ensemble model construction with Stacking

2.3 Experimental Setup

2.3.1 Experimental Protocol

In order to perform a comparative analysis of the different ensemble learning techniques, an experimental protocol has been developed, which is described in Figure 2.5. The main objective of this protocol is to examine the theoretical superiority of ensemble models over individual regression models, and compare the different ensemble learning techniques in the context of an indirect test strategy.

The experimental protocol therefore includes four distinct phases consisting of (i) population partition, (ii) feature selection, (iii) model construction, and (iv) test efficiency evaluation. The definition and the details of the different phases are explained hereafter.

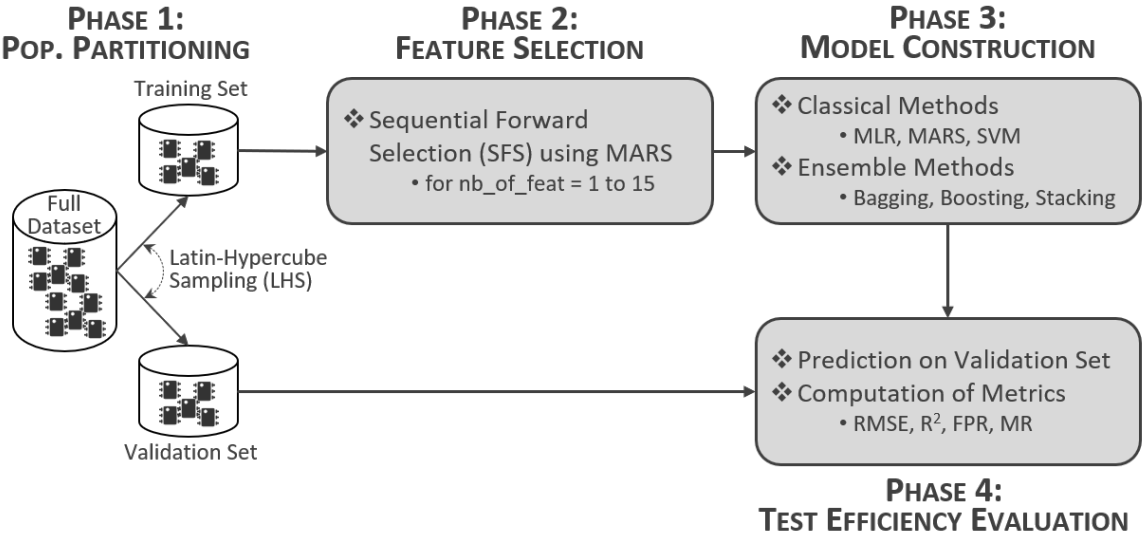


Figure 2.5: General overview of the experimental protocol

(i) Population Partition

In order to ensure a correct evaluation of the prediction model quality, it is essential to divide the dataset into training and validation sets. Instances of the training set will be used for the learning of the prediction model while the model evaluation will be conducted using the unseen instances of the validation set. Predicting unseen instances permits to verify that the prediction model built on the training instances avoids overfitting, thus guaranteeing its generalization capabilities.

Moreover, to efficiently evaluate the performance of the established regression model, the training and validation sets must have similar statistical properties, i.e. same distribution and standard deviation and they should reflect the expected device distribution and properties. One way to ensure consistency between both sets is to perform a stratified sampling from an initial full dataset which contains all the needed measurements to implement an indirect test strategy. We could achieve a stratified sampling with regard to one performance specification while using Latin Hyper-cube Sampling (LHS) described in [34] and introduced in [28] in the context of an indirect test strategy to divide the initial Learning Set into two identical sets in terms of their statistical properties.

Finally, it would be more advantageous to achieve a similar outcome when the sampling is performed on multiple variables in a multivariate distribution. This might resemble a case where multiple performance specifications are required to be predicted based on a model established using a homogeneous learning set. Therefore, we opted to use an extended version of the LHS, called conditioned LHS, described in [35].

(ii) Feature Selection

Generally, a large number of indirect measurements exists in the initial dataset. A necessary step is to select only a subset of relevant indirect measurements for use in the model construction, among all the available indirect measurements. This process is identified as feature selection in the domain of machine learning, where the selected indirect measurements are called features. Selecting only a limited number of features is essential in order to avoid the curse of dimensionality and enhance generalization ability by reducing over-fitting. Moreover in the context of the indirect test strategy where the objective is to reduce the overall test cost, it permits to maintain the simplicity of the prediction model and limit the number of measurements realized during the production testing phase. It also helps in providing insight about the circuit performance.

There are different types of feature selection algorithms, which can be divided into three categories, namely filters, wrappers, and embedded methods [20] as explained in Section 1.5. The most commonly used solution in the context of indirect test strategy is a wrapper method based on Sequential Forward Selection (SFS). The procedure starts by building a regression model for each available IM, and then selecting the IM that generates a model with the minimum prediction error (lowest *RMSE* score). At the second iteration, a regression model is built for each pair of IMs which 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 as explained in Section 1.5. In this experiment, we have implemented SFS using a MARS regression model, and limiting the selection to 15 features maximum.

(iii) Model Construction

Based on the features selected in the previous step, it is now possible to build regression models of different types. The main objective here is to investigate whether ensemble methods can outperform classical regression methods. Practically, we have decided to compare the three mostly used classical prediction models, namely MLR, MARS and SVM presented in Section 1.4, with some ensemble models. For the construction of the ensemble models, there are infinite possibilities and ways to build these models. We have chosen to study five ensemble models covering the different categories presented in Section 2.2, as described in the following.

- *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 (RandF) model as a fourth base model, obtained with a bagging algorithm applied on 300 decision trees. In both cases, the aggregation of the base learners is achieved by the MARS algorithm.

(iv) Test Efficiency Evaluation

Finally, the last phase of the experimental protocol 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. The various performance metrics that were discussed in Section 1.6 are then computed, namely the *NRMSE* which reflects the accuracy of the model, the R^2 score which illustrates the goodness-of-fit, and the *FPR* score which quantifies the reliability of the model. Moreover, if the performance specification test limits are available, the *MR* is also computed, which is an important metric in the context of an indirect test strategy since it gives an idea about the overall yield in the final production line.

2.3.2 Case Study

The test vehicle used in this experiment is a Low-Noise Amplifier (LNA) developed by NXP Semiconductors for which we have a production test data comprising 3,850 devices. More precisely, the test data includes conventional measurements of three different RF specification performances, namely the gain (G), the output power at 1dB compression point (P1dB), and the third-order intercept point (IP3). In addition, the test data also includes 79 low-cost indirect measurements, that correspond to DC voltages on internal nodes and additional DC signatures delivered by built-in process monitors.

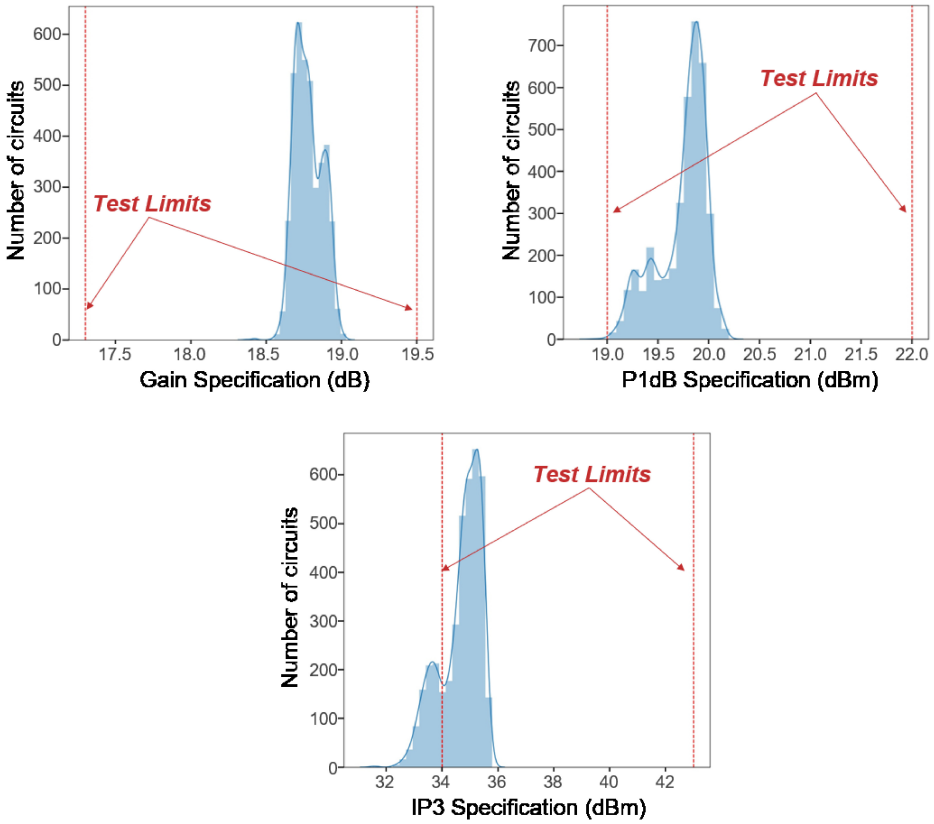


Figure 2.6: Distribution of the three RF specifications

Figure 2.6 illustrates the distribution of the three RF performances under investigation. It clearly appears that the three RF performances exhibit a non-Gaussian distribution. Moreover, the three RF performances correspond to three very different situations, as highlighted in Table 2.1 that summarizes the main statistical characteristics of the full dataset, for each RF performance. Indeed, it is obvious that there is a strong disparity between the three RF performances in terms of range, dispersion, and location of the distribution with respect to the test limits.

Table 2.1: Main characteristics of the three RF performances

	RF Performance		
	Gain	***P1dB***	***IP3***
Mean value	17.78dBm	19.74dBm	34.68dBm
Std deviation	0.09dB	0.24dB	0.72dB
Coef. of variation	0.49%	1.22%	2.08%
Meas. uncertainty	0.1dB	0.1dB	0.5dB
Test limits	[17.3dBm;19.5dBm]	[19dBm;22dBm]	[34dBm;43dBm]
# circuits within spec	3850	3847	3043
# circuits out of spec	0	3	807

For the gain performance, we observe a very tight distribution with dispersion of only 0.51% and a standard deviation of 0.09dB that is slightly smaller than the typical RF measurement uncertainty of 0.1dB. Such a situation is far from perfect, since we might be dealing with the noise present in the measurement setup, rather than modeling the impact of manufacturing process variations on the circuit performance. Furthermore, the test limits are located far away outside the distribution of available samples. As a consequence, there are no faulty circuits with respect to the gain performance, which means that the evaluation of the MR metric is pointless.

For the P1dB performance, we observe a slightly larger distribution with a dispersion around 1% and a standard deviation of 0.24dB that is a bit more than twice the typical RF measurement uncertainty of 0.1dB. The situation is therefore more favorable to capture the impact of manufacturing process variations on the circuit performance. However as for the gain performance, the distribution mostly falls within the test limits, even if the lower test limit is located very close to the left tail of the distribution. Only three circuits have a P1dB performance inferior to this limit, which constitutes a negligible portion of the population (less than 0.1%). The evaluation of the MR metric is therefore also meaningless for this performance.

Finally, for the IP3 performance, we observe a significantly larger distribution with a dispersion around 2% but a standard deviation of 0.72dB that is only about 1.5 times the typical RF measurement uncertainty of 0.5dB. The interesting point for this performance is that the lower test limit falls within the distribution of available circuits. In particular, 807 circuits exhibit an IP3 value inferior to this limit, which corresponds to around 20% of the population. Such a proportion is sufficient to allow the evaluation of the MR metric.

Taking into account this diversity, it is particularly interesting to see how the indirect test approach, and more specifically the different types of model, are able to handle these different situations. In this objective, the experimental protocol presented in the

previous subsection has been applied. The initial dataset has been partitioned into two independent sets, namely the Training Set (2000 instances) and the Validation Set (1850 instances), with the help of cLHS described in Section 2.3.1 in order to preserve its statistical characteristics and does not introduce any bias. As illustrated in Table 2.2, it can be observed that, for each RF performance, both sets exhibit similar characteristics in terms of coefficient of variation and proportion of good/faulty circuits.

Table 2.2: Statistical characteristics of the training and validation sets

		RF Performance		
		Gain	***P1dB***	***IP3***
Training Set 2000 instances	Coef. of Variation	0.48%	1.21%	2.08%
	# of good circuits	2000	1999	1591
	# of faulty circuits	0	1	409
Validation Set 1850 instances	Coef. of Variation	0.49%	1.23%	2.09%
	# of good circuits	1850	1848	1452
	# of faulty circuits	0	2	398

2.3.3 Initial Results

Prediction of Gain (G)

Figure 2.7 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 the model generated by MARS algorithm compared to MLR and SVM models. The best solution is actually obtained using a MARS model built with nine features, with an R^2 score of 0.65, an *NRMSE* score of 0.29% and an *FPR* score of 2.9%. Regarding ensemble methods, models generated using stacking are more performing than models generated using boosting or bagging methods. The best solution corresponds to an ensemble model built with nine features that combines MLR, MARS, and SVM models, with a Random Forest (RandF) ensemble model. This ensemble model permits to reach an R^2 score of 0.72, an *NRMSE* score of 0.26% and an *FPR* score of 1.5%.

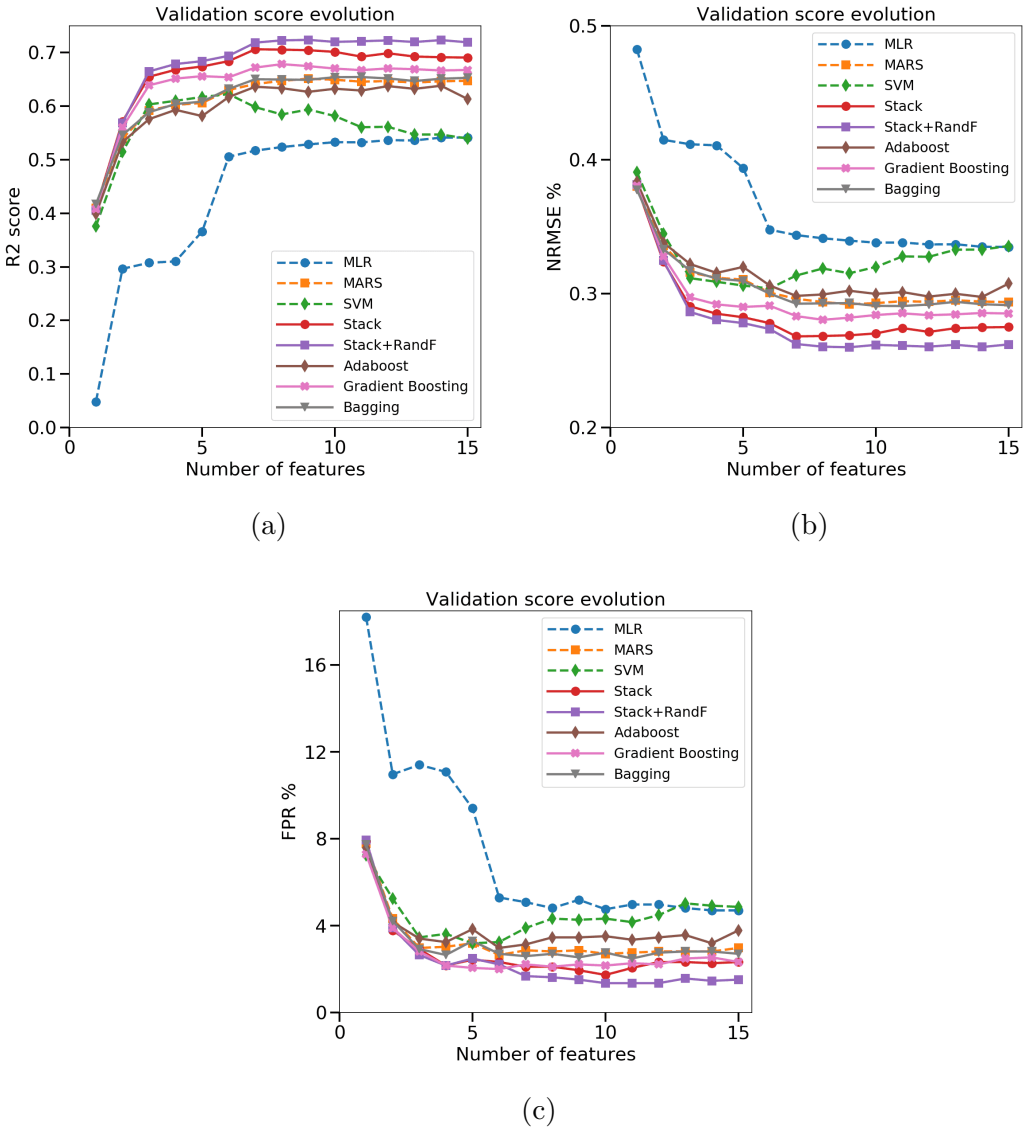


Figure 2.7: Comparison of classical and ensemble methods for gain prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score

More generally for the gain specification, these results show that it is possible to benefit from 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 an R^2 score that improves roughly by 10%, 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 an *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.

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

Figure 2.8 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.

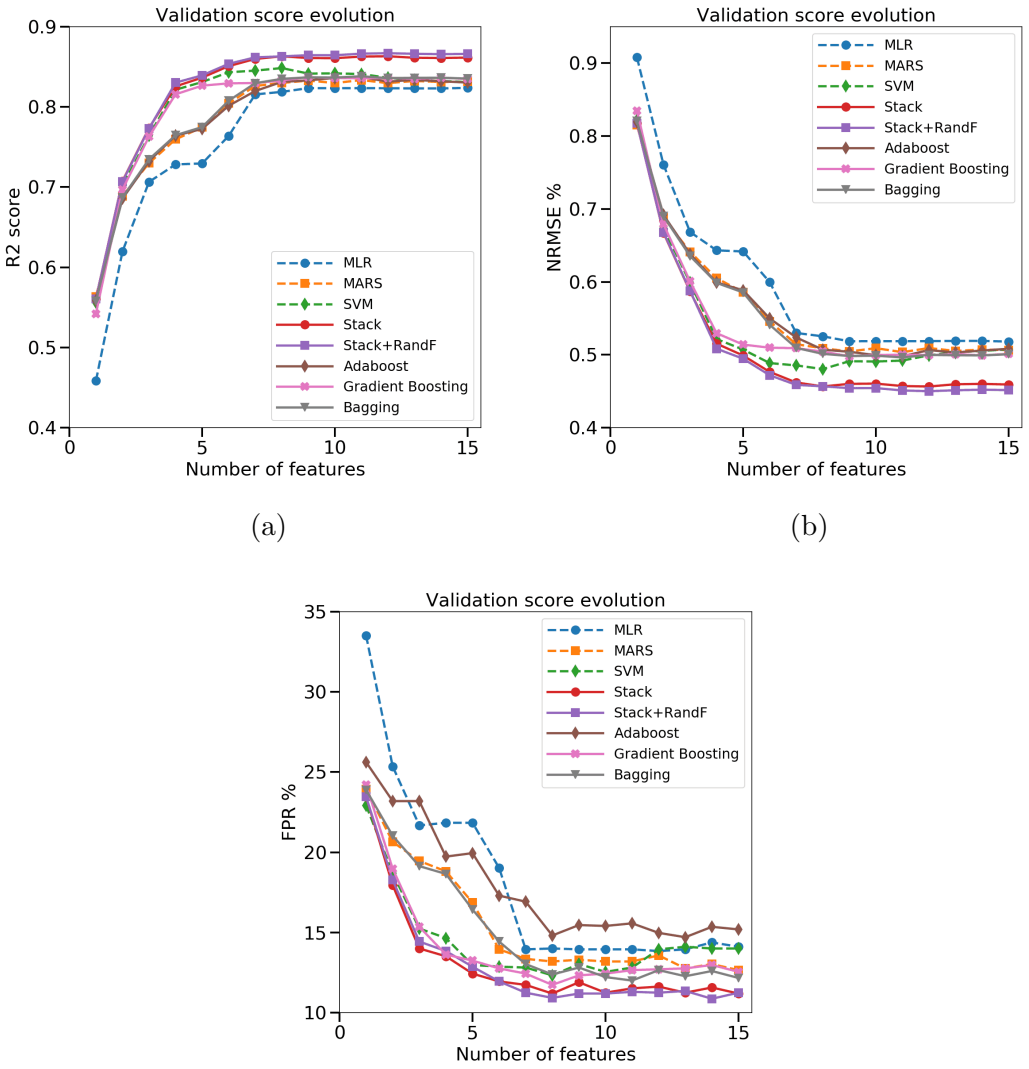


Figure 2.8: Comparison of classical and ensemble methods for P1dB prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score

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

Globally for the P1dB specification, there is a slight benefit in using ensemble models generated with stacking method 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 improves by roughly 2.3%, whereas both the *NRMSE* and *FPR* scores remain in the same range, thus a limited improvement is achieved in terms of model's accuracy and robustness. It should be noticed 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.

Prediction of third order intercept point (IP3)

Figure 2.9 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 the case of the IP3 specification, a similar behavior is observed as in the case of the P1dB specification, i.e. the most powerful model obtained with classical methods is SVM model and the most powerful models generated with ensemble methods are models generated with stacking method. 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 an SVM model built with 14 features that exhibits an R^2 score of 0.93, an *NRMSE* score of 0.57% and an *FPR* score of 0.59%, while the best solution obtained with an ensemble method is a stacked model built with 14 features that exhibits an R^2 score of 0.94, an *NRMSE* score of 0.52% and an *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. Note 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.

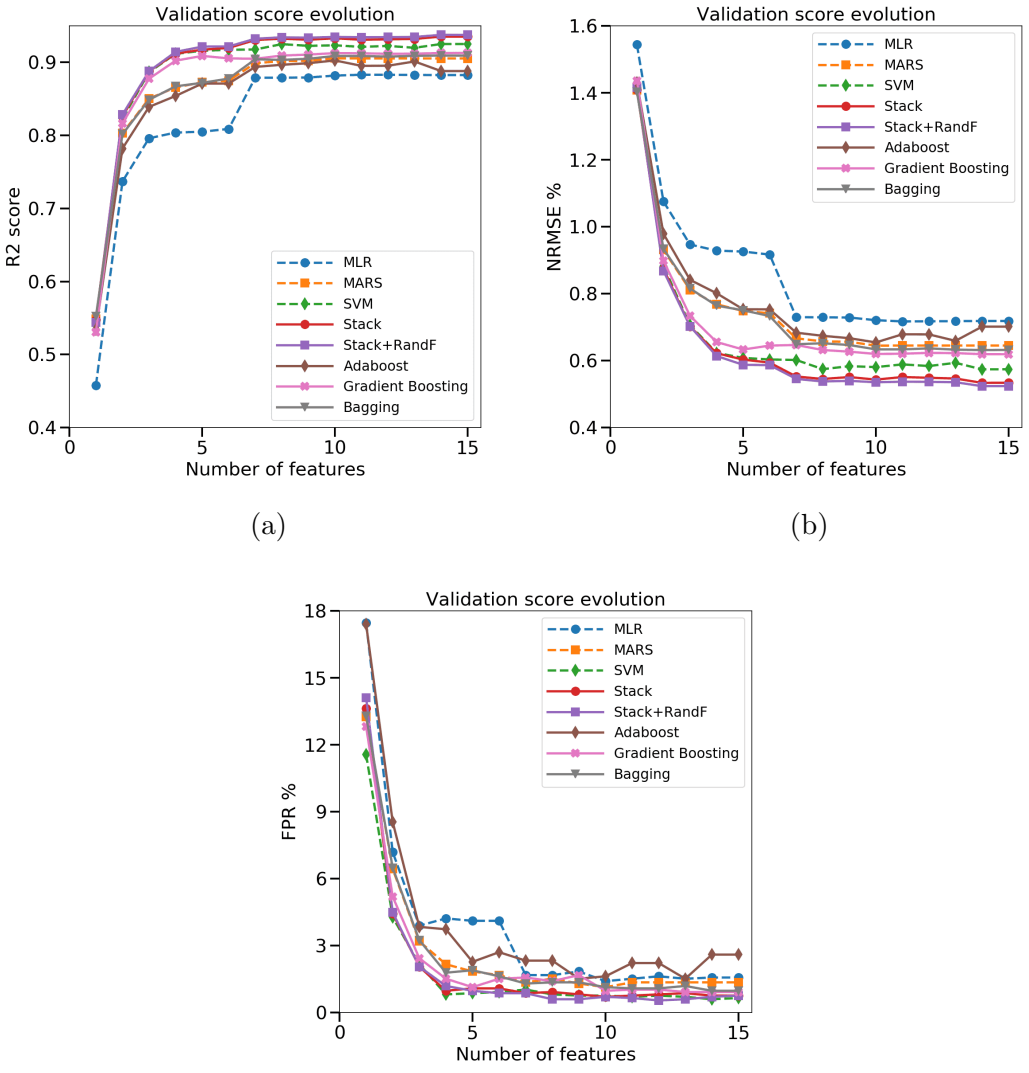


Figure 2.9: Comparison of classical and ensemble methods for IP3 prediction: (a) R^2 score, (b) *NRMSE* score and (c) *FPR* score

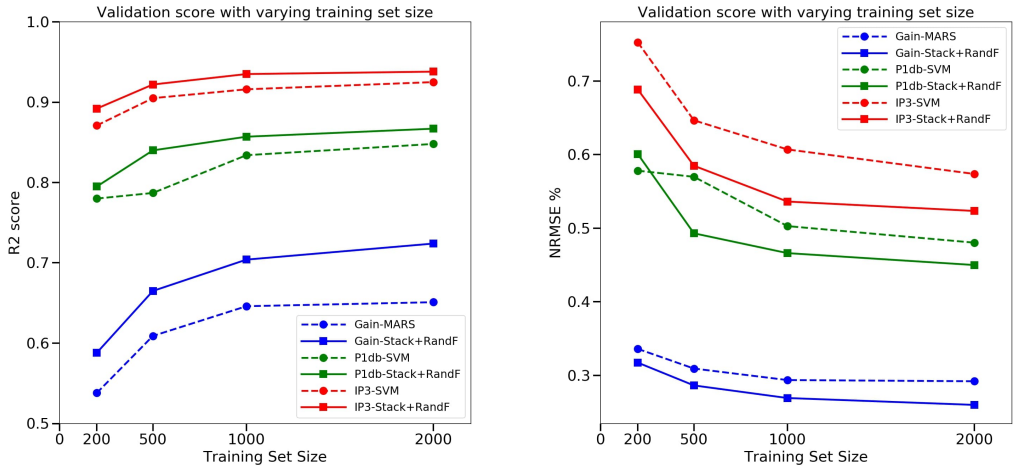
2.3.4 Influence of the training set size

Following our analysis of the initial results, we have decided to further extend the experiment, investigating whether the use of ensemble learning might offer additional benefit with respect to the training set size. Indeed, the main objective of using an indirect test strategy is to reduce the overall test cost, which could be eventually achieved by limiting the number of indirect measurements performed during the production testing phase, but also by decreasing the number of circuits used during the learning phase to build the regression models. Hence, it is important to study the quality achieved by both classical and ensemble models while varying the training set size.

In this objective, additional experiments have been performed gradually reducing the size of the training set. More precisely, we have considered four different sizes of training sets: 2,000, 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 cLHS, in order to preserve the distribution of each specification.

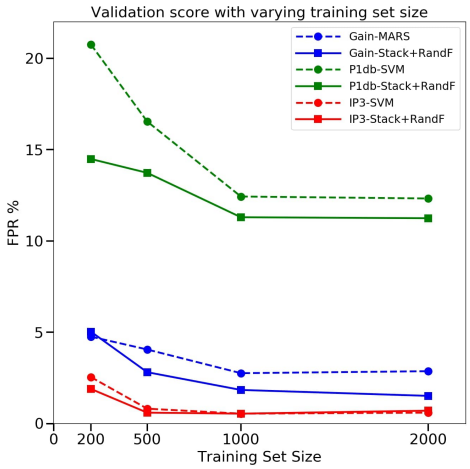
For each RF performance, we have selected the best ensemble model and the best classical model in terms of accuracy when the training is performed on the full initial training set of 2,000 circuits. In the case of ensemble learning, it is the same type of model that offers the best accuracy for the three performances, i.e. a stacked model built from four base learners including the Random Forest model. In contrast for classical models, the type of model that offers the best accuracy depends on the considered RF performance, i.e. a MARS model for the gain, and an SVM model for the two other RF performances. 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 the metrics are evaluated on the same validation set composed of 1,850 devices. Results are summarized in Figure 2.10, which shows the evolution of the different metrics with respect to the size of the training set used, for the three RF performances.

As expected, there is a global degradation in the achieved scores for both classical and ensemble models as the size of the training set is reduced. This degradation is almost negligible when the training set size is reduced from 2,000 to 1,000 devices but it is more pronounced when the training set size is further reduced down to 500 and 200 devices. Furthermore, the level of degradation differs from a situation to another, depending on the considered metric and the evaluated RF performance.



(a)

(b)



(c)

Figure 2.10: Influence of the training set size on performances achieved for the best classical and ensemble learning models for the 3 RF specifications: (a) R^2score, (b) $NRMSE$ score and (c) FPR score

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. However, the superiority of ensemble models does not increase as the training set size reduces. There is even an exception for the P1dB specification where 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 similar 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. Nonetheless, 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 observed of the ensemble model 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 stability with respect to the training set size. However, by referring again to Figure 2.10, it is fair to argue that the different scores achieved by using the classical models built on 2000 training samples can be reproduced or even improved by using ensemble models built only on 500 training samples. The only exception observed here is in the case of the P1dB prediction and concerns only the model robustness, where the *FPR* score of the ensemble model built on a reduced set of 500 training samples does not outperform the *FPR* score of the classical model built on the full training set of 2000 samples. Overall, the use of ensemble models instead of classical models could be justified, since it might permit to reduce the number of circuits to be used during the training phase.

2.4 Results Summary

Table 2.3 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. In particular, based on the results presented in Table 2.3, we can observe that whenever

a single model achieves a high R^2 score, the level of improvement gained by the use of ensemble methods decreases. The use of ensemble methods also permits to obtain an improvement in the accuracy of the generated models for the three specifications, however it is considered as a minor improvement with a reduction of the *NRMSE* score of only 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.

Table 2.3: Comparison between classical and ensemble methods: Summary of best results for the three 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%	0.7%	4.2%	14

Still, an important point to underline is that when using classical methods, the type of the model that gives the best results (MARS, SVM...), differs depending on the specification under investigation, which might hinder the implementation of the indirect test strategy, since we have to include a model type selection phase for each new product or scenario. In contrast, ensemble models built with stacking always lead to the best result. It is an interesting characteristic to have a solution able to handle a variety of different situations in a robust manner, while preserving its superiority over other regression model types.

Hence, globally, the use of ensemble models that are built using stacking appears to be an interesting and robust option. Moreover, it should be mentioned that we did not 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 diversify even more the collection of models, by trying another type for the aggregating model (MARS model in this study), or by exploring the use of multi-layer stacking ensemble methods.

Test efficiency evaluation

More generally, this study also opens the question on what is a pertinent metric to evaluate the indirect test efficiency. Indeed, the results show that the achieved performances can 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 an R^2 score of around 0.7, but attain a good accuracy with an *NRMSE* below 0.3%, and a 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 an R^2 score around 0.85, together with a good accuracy with an *NRMSE* smaller than 0.5%, but, however, 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 a good quality in terms of goodness-of-fit with an R^2 score higher than 0.9, a good accuracy with an *NRMSE* around 0.5%, and a good reliability with less than 1% of the devices that exhibit a prediction error that exceeds the classical measurement uncertainty.

Moreover, 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 the available samples. For instance, in the 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 within the specification limits, 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 high scores in terms of 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 2.11, 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.

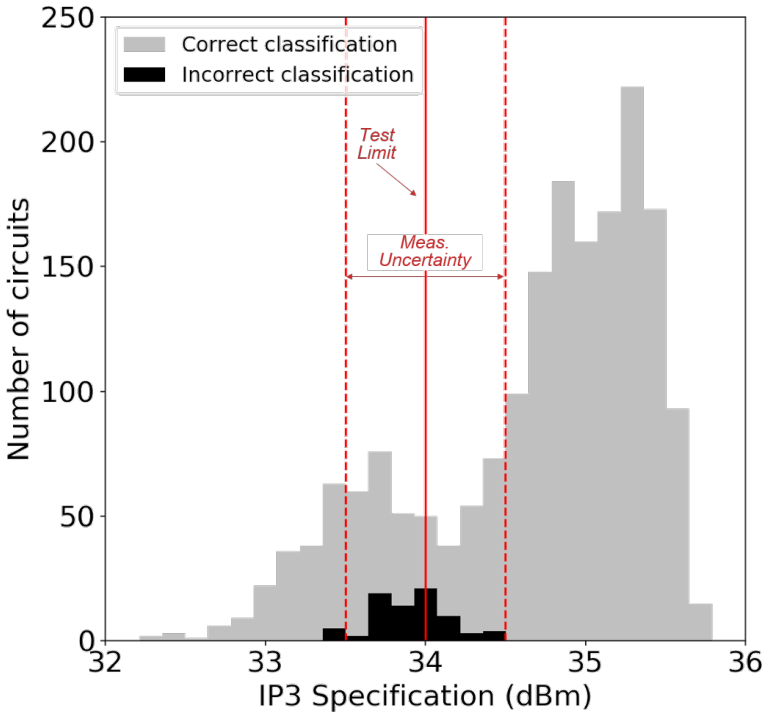


Figure 2.11: Illustration of misclassified devices by using the "Stack+RandF" ensemble model for IP3 specification

Trusted Misclassification Rate

To further explain this point, let us analyze more in details Figure 2.11. When the measurement uncertainty is taken into account, it exists a region of uncertainty around the test limit where the circuits might comply with the specification or not; only the circuits outside this region, can be trustfully classified as good or faulty when using 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 faulty circuits that do not comply with the specification, and 1,270 are trusted good circuits that comply with the specification. Now looking at the results of the implemented indirect test strategy, 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 reflect this situation in a clear manner.

The situation might be better evaluated or understood, if the classical misclassification rate is accompanied by a new metric that represents the percentage of circuits that have an incorrect decision with the indirect prediction among the number of circuits that have a certain decision with the conventional measurement. This new metric is

called Trusted Misclassification Rate (*TMR*). For our case study, 1450 circuits of the validation set are considered as trusted (i.e. outside the uncertainty region). Among them, only five circuits are wrongly classified with the indirect test strategy, which corresponds to a very low *TMR* of 0.34%. We believe that this metric can be legitimate and more representative of the intrinsic ability of a prediction model to correctly classify circuits under test. The actual misclassification rate achieved in production when using the indirect test will be somewhere between the classical misclassification rate and the trusted one.

2.5 Conclusion

In this chapter, we have investigated the benefit that could be achieved by using ensemble methods in the context of indirect test for RF circuits. Different ensemble methods based on bagging, boosting, and stacking have been studied and compared to classical individual models. The developed experimental protocol was applied to a practical case study, which was provided by a production test dataset from an LNA integrated circuit. According to this experiment, it seems that ensemble models built with stacking outperform the other 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, ensemble methods surpass the various classical individual models' performances, 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 further explorations, in particular by 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 chapter highlights a meaningful question in the context of indirect RF testing, which is the pertinence of the metrics that are usually considered to evaluate the quality of a model, and the level of confidence we can have through these metrics. 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. An additional metric, called the trusted misclassification rate, has been introduced that permits to evaluate the ability of a prediction model to perform correct classification while taking into account the conventional RF measurement uncertainty. However, this study actually pinpoints 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 faulty with respect to a given specification, based only on indirect measurements.

In the following chapter, a fallback scenario for the implementation of the indirect test strategy will be investigated based on the concept of trusted classification regions with the objective to increase the confidence level and offer the possibility to reduce the global testing costs without compromising the test quality.

Chapter 3

Adaptive Test Flow

3.1 Introduction

In order to implement an indirect test strategy in an industrial context, it is essential to preserve the test quality achieved by the conventional specification test. However, one of the main issues that today limits the wide deployment of the indirect test strategy in industry is a problem of confidence in the predicted results. Indeed, the machine-learning algorithms used to build regression models are perceived as a black box and often induce a lack of confidence. To cope with this issue, an extension of the indirect test strategy has been proposed, called the two-tier adaptive test flow. The objective of this approach is to preserve the test quality of specification testing while leveraging the low-cost of indirect testing.

In this chapter, we explore a novel implementation of this approach in the context of prediction-oriented indirect test. This chapter is organized as follows. Section 3.2 introduces the principle of the two-tier adaptive test flow, briefly reviews the state-of-the-art, and presents the proposed implementation. Section 3.3 is then dedicated to the experimental protocol developed in order to assist the test engineer in the elaboration of the two-tier adaptive test flow. Finally, the case study and results are presented and discussed in Sections 3.4 and 3.5, respectively.

3.2 Two-Tier Adaptive Test-Flow

3.2.1 Principle

The principle of the two-tier adaptive test flow is illustrated in Figure 3.1. As in the classical indirect test implementation, it involves two distinct phases: the training and production testing phases. The training phase is identical to the classical implementation, only the production testing phase differs. More precisely, the idea is first to process every device by the indirect test, i.e. to predict its performances based only

on the low-cost indirect measurements using the models learned in the initial training phase. However, the predicted values are also accompanied by an information on the confidence in the predictions. If the confidence is high enough, predictions are considered reliable and the device is labeled according to the indirect test decision. This constitutes the first tier. If the confidence in the prediction is insufficient, the device is then directed to a second tier in which it is submitted to a standard specification test, i.e. the conventional RF measurements are performed and the device is labeled according to these measurements. The underlying assumption behind this approach is that the large majority of devices will be sorted by the first tier, and that only a small fraction of devices need to go to the second tier. This approach then permits to maintain a significant test cost reduction, but offers more confidence in the achieved test quality.

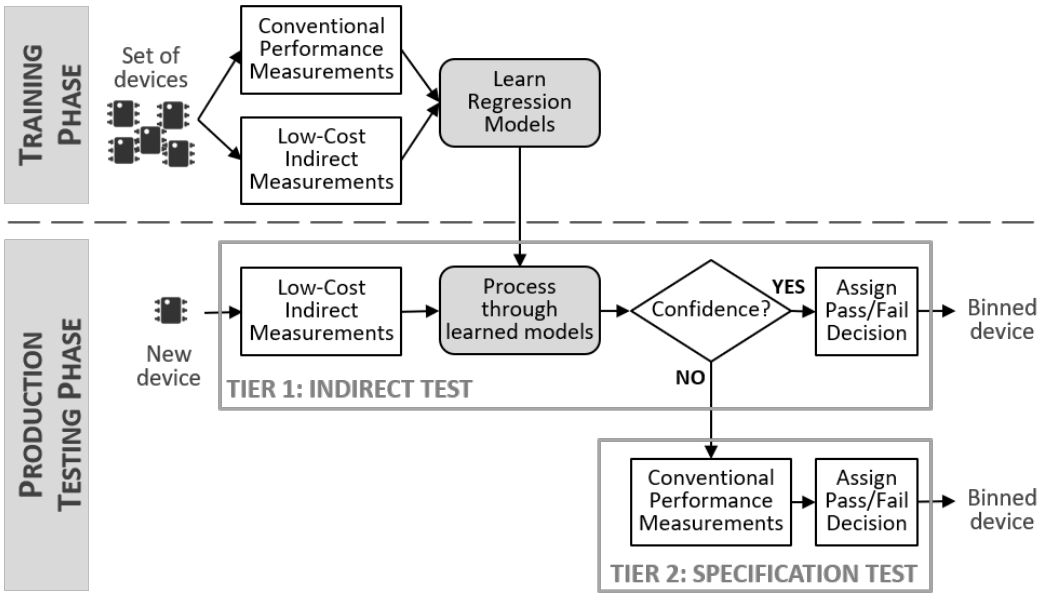


Figure 3.1: Two-Tier Adaptive test flow synopsis

3.2.2 State-of-the-art on adaptive indirect test

Traditionally, test content, test flow and test limits are statically set, which means that all parts are tested in the same way, regardless of their individual performances. On the other hand, in adaptive testing, test content, test flow or test limits can be changed for each part based on manufacturing test data or statistical data analysis [36]. This concept has emerged in the early nineties to optimize the tests applied on VLSI digital dies [37]. It has then been largely exploited in the context of Iddq-testing of digital circuits [38]. More recently, adaptive approaches have been explored for analog and mixed-signal circuits [39]. In this section, we focus more specifically on adaptive testing in the context of indirect testing for analog/RF ICs.

Classification-oriented indirect test

Adopting an adaptive test flow to improve the quality and accuracy of an indirect test strategy was firstly introduced in [7]. The authors have implemented an ontogenic neural network to create two guard-bands between faulty and good circuits in the indirect measurement space. During the production testing phase, each new device first goes through the indirect test tier. If the device falls outside the guard-bands, it is assigned to the dominant class (good or faulty) based solely on the low-cost indirect measurements. Oppositely, if the device falls within the constructed guard-bands, it is directed to the second tier where it is retested using the conventional specification test.

In another work, the authors of [40] have proposed a confidence estimation, by creating three different regions in the indirect measurement space, where the prediction of new devices can be trusted, discarded, or retested. The boundaries of these different regions are established using an SVM classifier that differ between two different classes: trusted and untrusted predictions.

Prediction-oriented indirect test

The adaptive two-tier test approach has also been explored in the context of prediction-oriented indirect test. It was firstly introduced in [41], where the authors proposed a strategy based on model redundancy. The main idea is to apply various prediction models on the same training set, while using a different set of indirect measurements for each model. Whenever a lack of consistency in the different predictions for a particular instance is detected, the circuit is then re-directed for further thorough testing.

In an another work, the authors of [42] have proposed two adjustable defect filters, in order to avoid the entailed risks in predicting the performance of marginal and extreme instances. The first one is a strict filter, where all the instances that are adequately represented in the training data will have their performance predicted by using the established regression function. On the other hand, all the suspicious instances are redirected to the more lenient filter, which will discard gross defects (extreme instances) and re-test marginal devices in a classical manner. Unlike the introduced guard-bands in [7], where a classifier is trained to create a buffer zone in the indirect measurement space, the defect filters are constructed based on Kernel Density Estimation of the indirect measurements, which has been introduced in [43].

Multi-site indirect test

Finally, in the case of multi-site testing, the authors of [44] have proposed an implementation of an adaptive test strategy to reduce the test time. The main supposition is that the indirect test strategy, which has already defined the set of the most pertinent performance indicators, is capable of replacing the specification based testing and can replicate its accuracy. The adaptive element in this strategy is the number of indirect

measurements included in the prediction model: the less are incorporated, the more time can be saved. Thus, the learning phase involves the ordering of the available features and the incremental training of different regression models by adding a new feature at each iteration. During the production test, the test program starts with an evaluation using a model with a low number of indirect measurements. If all the sites are predicted with an acceptable level of confidence, the circuits under test do not have to keep on exploring the remaining indirect measurements and the test program halts, leading to test time improvement.

3.2.3 Proposed Solution

As discussed in the previous section, there are diverse ways of approaching an adaptive indirect test implementation. Our target is to develop an adaptive solution in the context of a prediction-oriented indirect test. Compared to the solution proposed in [42] where circuits are re-directed based on their distribution in the indirect measurement space, our preference is to base the adaptive strategy on the distribution of the predicted instances in the RF performance space. Our intention is also to limit as much as possible the number of used indirect measurements in order to maximize the cost reduction. Hence, the solution proposed in [41] is not suitable, since it relies on the building of redundant models that involves different indirect measurements, which necessarily increases the number of required indirect measurements.

In this section, we present a novel implementation of the two-tier adaptive test flow in the context of prediction-oriented indirect test. The idea is to evaluate confidence based on a tolerance zone around test limits. Indeed, experiments realized in Chapter 2 have shown that almost all of misclassified circuits are circuits with a predicted value close to a test limit, while correct decisions are taken for circuits with a predicted value far from test limits. Therefore, the proposal is to establish confidence by looking at the location of the predicted value with respect to a tolerance zone defined around a test limit.

This principle of confidence estimation is illustrated in Figure 3.2. In the proposed solution, any device with a performance prediction that falls outside the tolerance zone will be directly classified according to the indirect test prediction outcome with regard to the test limit, while any device with a prediction that falls within the tolerance zone will be directed to the second tier in order to be evaluated and classified through the conventional specification tests.

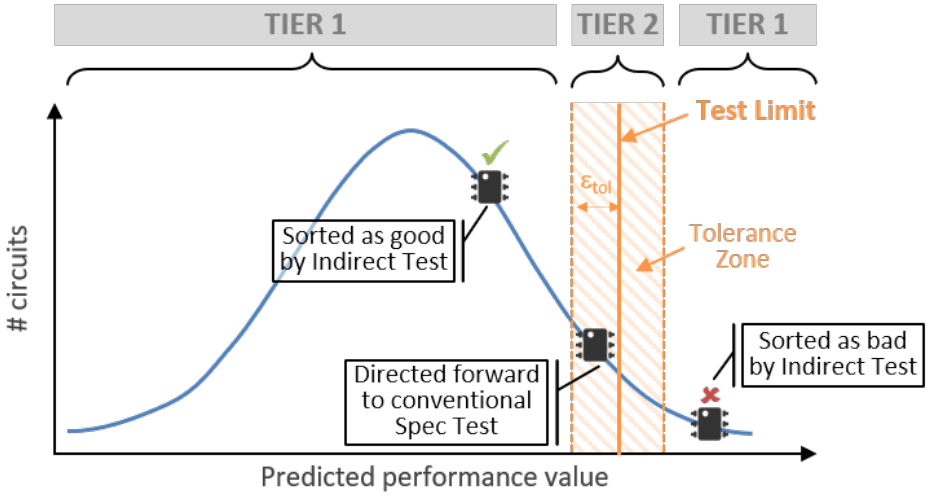


Figure 3.2: Principle of confidence estimation in the proposed two-tier adaptive test flow.

A chief interest of this solution is its simplicity while it has the ability to adapt to different industrial constraints. For this, the size of the tolerance zone established around the test limit is a crucial parameter. The first obvious solution would be to choose the size of the tolerance zone according to the conventional performance measurement uncertainty. Indeed, as discussed in the previous chapter, only circuits outside this zone can be trustfully classified as good or not. However, the size of the tolerance zone is an interesting parameter to exploit. Indeed, varying the size of the tolerance zone around the test limit permits to explore different trade-offs between test quality and test cost, which facilitates the development of a cost-effective test plan based on the two-tier adaptive test flow. More precisely, the initial implementation of the indirect test strategy developed in Chapter 2 did not include a tolerance zone around the test limit. Therefore, 100% of the evaluated devices during production test are processed by the low-cost first tier. The test cost is in this case minimum, but the test quality expressed in terms of misclassification rate might not be sufficient to meet with the 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, though at the expense of retesting a number of devices and therefore lower the benefit of using an indirect test strategy in terms of test cost reduction. It is therefore essential to have an appropriate setting of this parameter during the initial learning phase, depending on the targeted industrial constraints, in order to really benefit from the two-tier adaptive test approach.

3.3 Experimental Protocol

In the previous section, we have introduced the principle of the two-tier adaptive test approach. The practical implementation of such a principle implies several choices, such as the selection of pertinent IMs, the choice of the regression algorithm and the size of the tolerance zone. Obviously the achieved test quality relies on these choices. In this section, we describe the methodology that has been defined in order to assist the test engineer in the elaboration of the test flow. The general overview of this methodology is depicted in Figure 3.3.

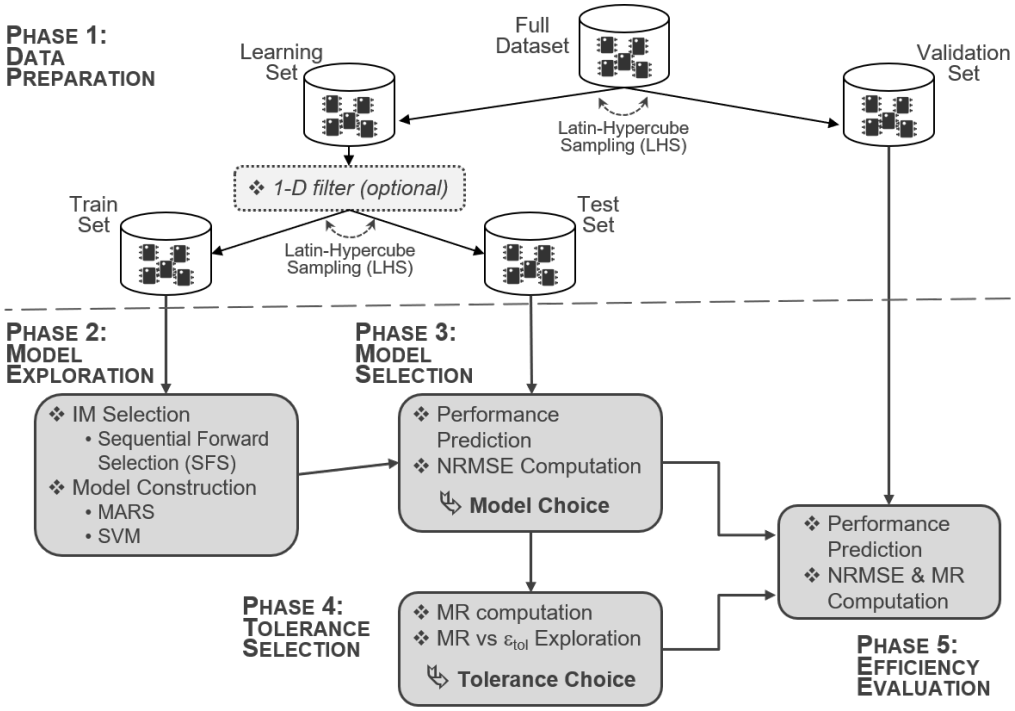


Figure 3.3: Principle of confidence estimation in the proposed two-tier adaptive test flow.

Data Preparation

Similar to the protocol developed in Chapter 2, the first phase concerns the data pre-processing and data preparation. The initial dataset should contain the conventional performance measurements and a large variety of indirect measurements for a sufficient number of circuits (typically several thousands). This full dataset is first partitioned into two datasets, called learning and validation sets. The learning set will be used to explore the different possibilities regarding the test flow implementation and to identify the best options. The second set is dedicated to the validation of the retained options using an independent set of devices; it is intended to represent the production testing phase. Note that, although both sets are independent, it is essential that they present

similar characteristics to ascertain the validity of the results. Therefore, the partitioning is realized using conditioned Latin-Hypercube Sampling (LHS), which is a sampling approach (presented in Section 2.3.1) that preserves the statistical characteristics of the initial distribution in the sampled sub-datasets.

The learning set is in turn partitioned into two subsets, i.e. the train set and the test set. The first one will be used to train the prediction models and the second one to evaluate the accuracy of the constructed models. It is important to perform this evaluation on different instances than the ones used for training in order to verify the model generalization ability, and avoid issues related to over-fitting.

Finally, note that it is often recommended in the literature [43, 45] to work with a dataset that does not contain extreme instances when implementing an indirect test strategy, due to their non-statistical nature. Indeed, since those random instances do not represent the actual distribution and are mainly caused by production defects, data outliers can spoil and mislead the training process, resulting in longer training times, less accurate models and ultimately poorer results. Consequently, an optional pre-processing step is implemented to exclude these instances from the learning set by applying an iterative one-dimensional filter, which have been proposed in [46]. The basic principle of this filter is, for a given parameter, to remove all instances that have a measured value outside the interval $[\mu - k\sigma; \mu + k\sigma]$, where μ is the mean value of the population for the considered parameter, σ the standard deviation, and k a positive integer that permits to choose the strictness of the filter. The filter is applied individually on each RF performance and each indirect measurement. The final list of circuits excluded by the filtering process is the union of all circuits pruned by the filter over all RF performances and indirect measurements.

Model Exploration

The second phase of the methodology is the model exploration. In this phase, a number of regression models will be built using different subsets of IMs. As seen in Chapter 2, the problem of selecting a pertinent subset of IMs within a large set of candidates remains a crucial step in the proposed framework. Similar to Chapter 2, the implemented approach is a wrapper method called Sequential Forward Selection (SFS). For this study, we have implemented such a procedure using MARS, and limiting the number of selected IMs to 15. The next step is then to train regression models using the selected IMs. Many different algorithms exist to perform this task. Classical algorithms, including Multiple Linear Regression (MLR), Multi- Adaptive Regression Splines (MARS), Support Vector Machine (SVM), or more elaborated algorithms that combine several models in an approach called ensemble learning, have been described in the previous chapters. For this study, our focus is mainly to explore the effects of implementing a two-tier adaptive test flow, and not to perform a complete comparison of the different model types. Therefore, we have implemented only two types of regression

models which are the most commonly used algorithms in the context of an indirect test strategy, i.e. MARS and SVM, for their capabilities of depicting non-linear behaviors.

Model Selection

The third phase of the methodology concerns model selection. In this phase, all the models learned in the previous phase are used to predict the performance specification of the devices included in the test set. The accuracy of these models is evaluated in terms of Normalized Root Mean Square Error (*NRMSE*), which is a normalized measure of the *RMS* prediction error expressed in percentage. Models with the lowest *NRMSE* are then retained as the best solutions for each performance specification.

Tolerance Selection

The following phase is specific to the implementation of a two-tier adaptive test flow. It is related to the exploration of the trade-off that can be achieved between the test quality, expressed in terms of Misclassification Rate (*MR*), and the test cost, expressed in terms of the percentage of devices that are retested by the conventional specification test. Practically, for each selected model, the misclassification rate is first computed with a tolerance zone set to zero (only indirect test). The size of the tolerance zone is then progressively enlarged in order to study the evolution of the misclassification rate versus the number of devices directed to the second tier. The appropriate size of the tolerance zone can be chosen for each RF performance with respect to a targeted test quality, i.e. the smallest size that does not overcome a predefined maximum *MR*.

Efficiency Evaluation

Finally, the last phase of the methodology is dedicated to the evaluation of the two-tier adaptive test flow efficiency. All the options retained in the learning phase are evaluated on the devices of the validation set. Indeed, it is important to verify that the efficiency established on the test set during the learning phase is preserved on the validation set, which is intended to be representative of the realistic conditions encountered during the industrial testing phase.

3.4 Case Study

3.4.1 RF Product

The case study used in this chapter is a front-end integrated circuit designed for Wireless Local Area Network (WLAN) applications. The three main specifications to be verified are the gain of the receiver chain (Rx-gain), the gain of the transmitter chain (Tx-gain) and the Error Vector Magnitude of the transmitter chain (Tx-EVM). The low-cost indirect measurements investigated for this product include standard DC

measurements performed on external nodes of the device, together with internal DC measurements (the device is equipped with an internal DC bus and internal probes that give access to the voltage at some specific nodes and signatures delivered by built-in process monitors). Overall, we have a total of 131 possible indirect measurements.

3.4.2 Measurement Campaign

In order to build an extensive dataset, a campaign of measurements has been carried out in a production test environment. Test data have been collected on more than 26,700 circuits, extracted from different wafers, which are purposefully fabricated under various process conditions including extreme process conditions. The test data, which include both the conventional measurements of the three RF specification performances and the 131 indirect measurements, constitute the full dataset. The main characteristics of this dataset are summarized in Table 3.1 for the three RF performances.

Table 3.1: Characteristics of the full dataset for the three RF performances

		RF Performance		
		Tx-EVM	***Tx-Gain***	***Rx-Gain***
Full Dataset 26,706 instances	Coef. of Variation	11.1%	3.0%	3.7%
	% of good circuits	76.4%	97.7%	100%
	% of faulty circuits	23.7%	2.3%	0%

It can be noticed that the characteristics of the population significantly differ depending on the considered RF performance. For the Tx-EVM, we observe a quite large distribution with a dispersion around 11%; more than 76% of the circuits satisfy the targeted EVM requirement. For the Tx-gain, the distribution is tighter with a dispersion of only 3%; almost 98% of the circuits satisfy the targeted gain requirement. Finally, for the Rx-gain, we also observe a tight distribution with a dispersion around 3.7%; in this case the targeted requirement is sufficiently far away from the distribution so that 100% of the circuits satisfy the requirement. At this point, it is important to underline that a number of wafers are manufactured with corner process conditions, and their circuits have been included in the population on purpose. Therefore, the proportion of faulty circuits is not representative of what would be the actual production yield under normal process conditions.

3.4.3 Data Preparation

Dataset Partitioning

The first step of the data preparation is the partition of the full dataset in two sets, i.e. the learning and validation sets. In this work, we choose an equal repartition between the two sets, realized using conditioned Latin Hypercube Sampling (c-LHS). The full dataset of 26,706 devices has therefore been partitioned into two sets of about 13,350 devices. The main characteristics of these two sets are summarized in Table 3.2, which provides the coefficient of variation, the percentage of good circuits and the percentage of faulty circuits, for the three RF performances.

Table 3.2: Main characteristics of the learning and validation sets for the three RF performances

		RF Performance		
		Tx-EVM	***Tx-Gain***	***Rx-Gain***
Learning Set 13,354 instances	Coef. of Variation	11.0%	3.0%	3.7%
	% of good circuits	77.2%	97.9%	100%
	% of faulty circuits	22.8%	2.1%	0%
Validation Set 13,352 instances	Coef. of Variation	11.3%	3.0%	3.7%
	% of good circuits	75.5%	97.6%	100%
	% of faulty circuits	24.5%	2.4%	0%

From this table, it clearly appears that the learning and validation sets exhibit similar characteristics in terms of distribution dispersion, and proportion of good or faulty circuits for each RF performance. It is thus confirmed that the use of conditioned Latin Hypercube Sampling permits to obtain several subsets with the same distribution characteristics as the initial population.

Use of the Optional Filter

The influence of the use of a filter during the learning phase has also been examined. In particular, a detailed analysis of the distribution and the properties of the circuits identified by the filtering process has been realized, in terms of number of circuits, repartition of these circuits with respect to their compliance with the RF specifications, and distribution of these circuits within the different subsets generated from the full dataset. Although this optional filter would be applied only on the learning set in the proposed methodology (Figure 3.3), the preliminary analysis presented here has been performed on the full dataset and varying the strictness of the filter, i.e. varying the value of k. Concretely, we have considered two different filters, i.e. a strict one with a limit at $\pm 6\sigma$ and a more relaxed filter with a limit at $\pm 10\sigma$ (the influence of these two filters on the achieved test efficiency will be studied in the following sections). For the completeness of the current preliminary analysis, we have also included here

more lenient filters, i.e. filters with a higher value of k (25, 50 and 100). Results are presented hereafter.

Figure 3.4 shows the evolution of the number of circuits identified by the filtering process with respect to the filter severity (note the log scale on the y-axis). As expected, the number of circuits with outlying values quickly diminishes as the filter becomes more relaxed. For the strict filter, 10,186 circuits are identified by the filtering process, which corresponds to 38.1% of the total population. This number reduces to 2,673 circuits for the relaxed filter, which corresponds to 10%. This number then rapidly falls down below 150 circuits for more lenient filters, which corresponds to a negligible portion of the population (less than 0.5%).

An important remark is that, whatever the filter severity, all circuits identified by the filtering process have extreme values because of one or several indirect measurements, but none because of the RF performances. Another remark is that, even for an extremely lenient filter with of value of ($k = 100$, 8 circuits present extreme IM values. Even small (only 0.03% of the population), this number is unexpected taking into account that none of the circuits present in the dataset has an RF performance value outside $\pm 6\sigma$ of the distribution.

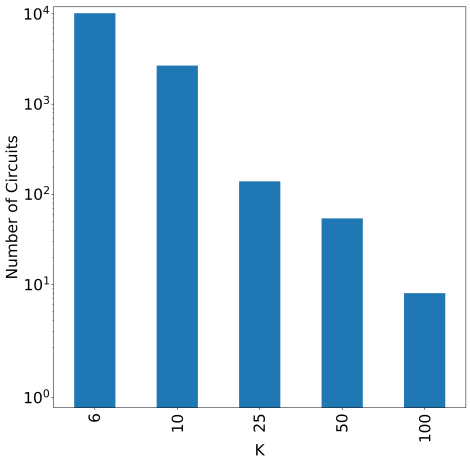
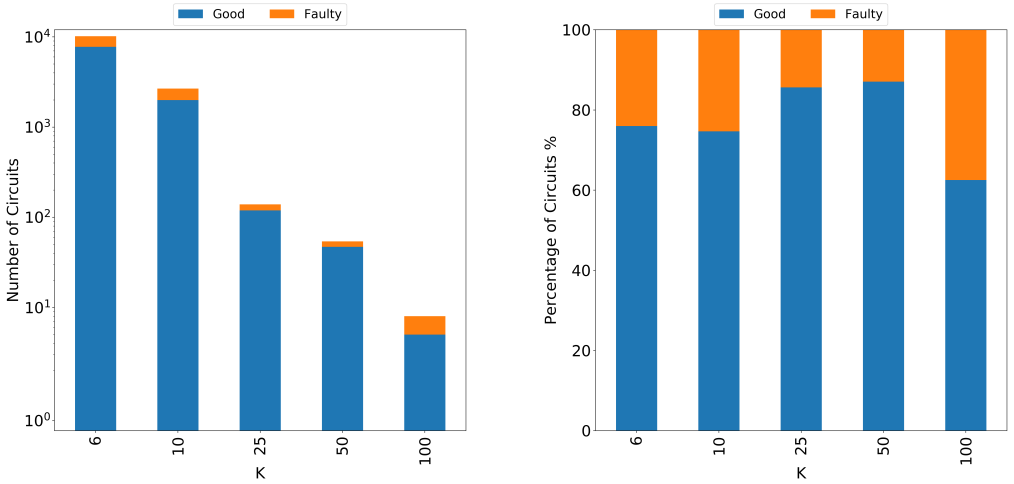


Figure 3.4: Number of circuits identified by the filtering process vs. filter severity

Then we have analyzed the repartition of circuits identified by the filtering process with respect to their compliance with the RF specifications. Results are reported in Figure 3.5. More precisely, Figure 3.5.a illustrates the number of good and faulty circuits within the set of circuits identified by the filtering process and Figure 3.5.b expressed this information in terms of percentage.

This figure shows that the proportion of faulty circuits among the total number of circuits identified by the filtering process varies between 13% and 37% depending on the filter severity. Globally, this proportion is in relative good agreement with the proportion of faulty circuits within the full dataset, i.e. 24%. However surprisingly, we observe a non-monotonic variation. Indeed, the proportion of faulty circuits among the total number of circuits identified by the filtering process is around 25% for the strict and relaxed filters with k=6 and 10. This proportion falls down to 13% for the lenient filters with k=25 and 50. It then increases up to 37% for the extremely lenient filter with k=100. This reveals that there is no direct relation between the fact that a circuit exhibits extreme values for indirect measurements, and the fact that it is a good or faulty circuit with respect to its RF performances. This point is important because it indicates that the use of a one-dimensional filter applied on the IMs during the production testing phase would be totally ineffective since it does not help to discriminate between good and faulty circuits. Even worse, it would eliminate a number of good circuits, provoking yield loss.

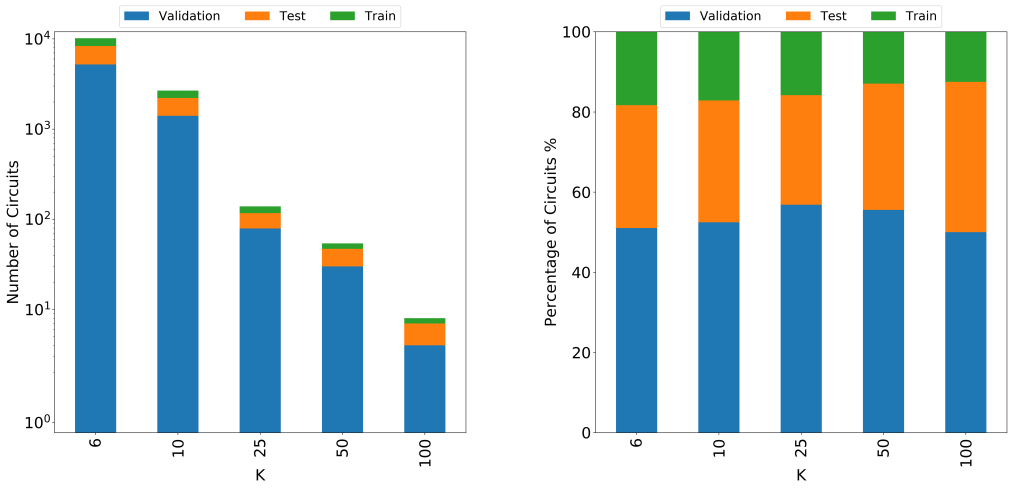


(a) (b)

Figure 3.5: Representation of filtered circuits in terms of good and faulty circuits

Finally, we have analyzed the repartition of the circuits identified by the filtering process within the different subsets used in the learning and validation phases (Train, Test, and Validation). Indeed, the conditioned-LHS process used for the partitioning of the population is realized considering only the RF performances. To ensure that there is no bias coming from this partition, it is essential to verify that the generated subsets also reflect the original dataset with respect to the indirect measurements. In particular, the proportion of circuits identified by the filtering process in each subset should be in accordance with the realized partitions, i.e. 18% in the train set, 32% in the test set, and 50% in the validation set. Results are reported in Figure 3.6. This figure

shows that, even if the number of circuits identified by the filtering process decreases as the filter becomes more relaxed, the expected repartition is globally maintained.



(a) (b)

Figure 3.6: Repartition of filtered circuits over Train, Test and Validation subsets

To summarize, this analysis shows that the use of a one-dimensional filter does not significantly modify the characteristics of the initial population with respect to the RF performances. Moreover, the circuits flagged by the filter are distributed in the different subsets in accordance with the realized partitions. It is therefore founded to investigate whether the use of such a filter can improve the accuracy of the models constructed during the learning phase, which might result in a better test efficiency during the production testing phase.

In the remaining of the chapter, we will consider only the strict and relaxed filters (k=6 and k=10) and we will study how the use of these filters during the learning phase impacts the test efficiency achieved on the validation set. For the sake of clarity, we stress that in the proposed methodology the optional filter is applied only to the learning set (i.e. train and test sets) and not to the validation set, which should remain representative of a realistic production population. The main characteristics of the filtered learning sets are summarized in Table 3.3. It can be clearly observed that, for each RF performance, the filtered learning sets exhibit a similar dispersion than the original learning set (maximum difference of 0.4%), and the proportion of good or faulty circuits is globally preserved (maximum difference of 1.2%).

Table 3.3: Characteristics of the filtered learning sets for the three RF performances

		RF Performance		
		Tx-EVM	***Tx-Gain***	***Rx-Gain***
10σ-filtered Learning Set 12,067 instances	Coef. of Variation	11.0%	2.9%	3.7%
	% of good circuits	77.4%	98.5%	100%
	% of faulty circuits	22.6%	1.5%	0%
6σ-filtered Learning Set 8,295 instances	Coef. of Variation	11.1%	2.6%	3.7%
	% of good circuits	77.8%	99.1%	100%
	% of faulty circuits	22.2%	0.9%	0%

3.5 Results

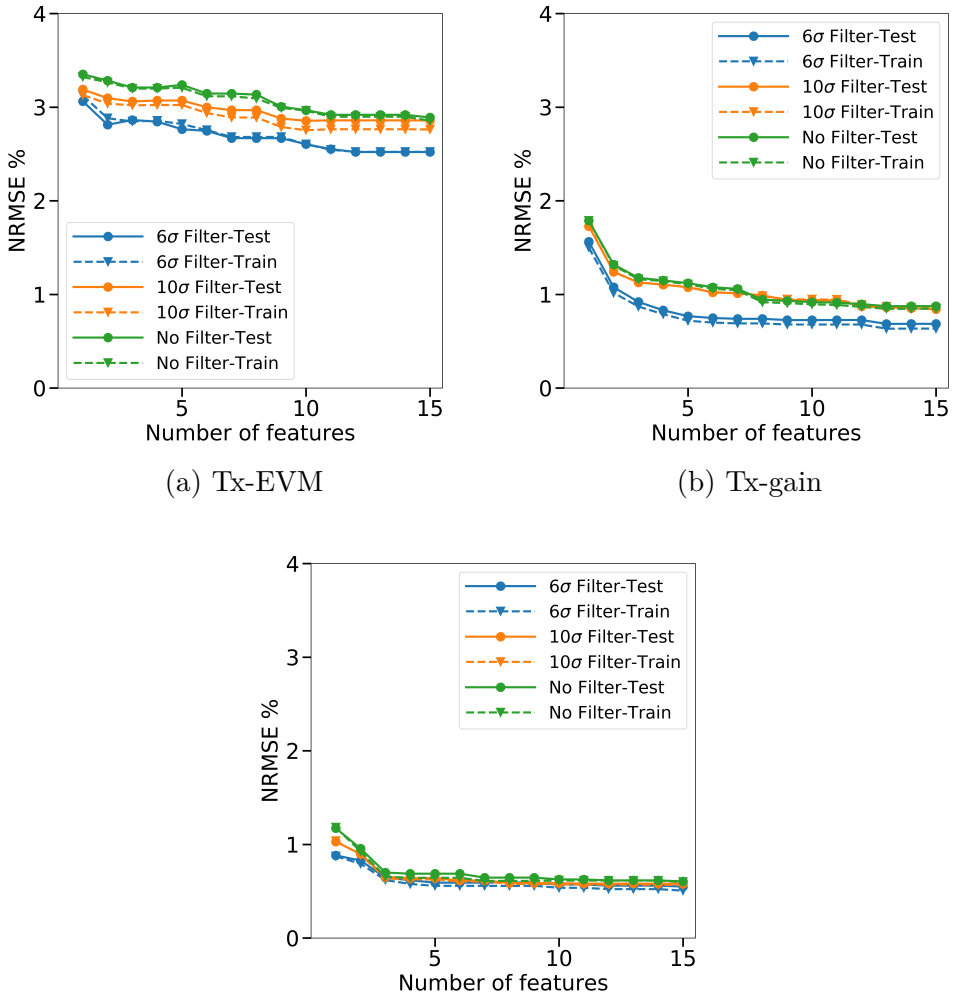
The methodology presented in Section 3.3 has been applied to our case study and the experimental results are discussed in this section. Model selection is firstly discussed and explained, then we examine the efficiency of a classical indirect test implementation, and finally we analyze the efficiency of the previously proposed two-tier adaptive test flow solution.

3.5.1 Model selection

The prediction accuracy achieved for the three different RF specifications is represented in terms of their *NRMSE* score, depicted in Figure 3.7 and Figure 3.8, for MARS and SVM models respectively. Moreover, we represent the evolution of the models' accuracy with respect to the number of IMs used to construct the prediction model, considering either the original or filtered learning sets.

By analyzing these results, several comments can be drawn. A first general comment is that the different regression models built for each RF specification do not suffer from over-fitting since there is no strong discrepancy between the *NRMSE* scores evaluated on train and test sets, for both types of regression models. Nevertheless, a slight advantage can be observed on this point for MARS models compared to SVM models. A second comment is that, whatever the model type, the level of accuracy differs over the different RF performances. Indeed, an *NRMSE* score below 1% can be achieved for the Tx-gain and Rx-gain performances, for both model types. The *NRMSE* score remains significantly higher for the Tx-EVM, with best score around 2.5% in case of a MARS model and around 1.6% in case of a SVM model. Finally, the last comment concerns the influence of the learning population. Its impact is mostly visible on the prediction of the Tx-EVM performance. For both model types, we observe that the use of a filter leads to an improvement in the accuracy of the constructed models, especially in the case of a strict filter.

From this exploratory phase, the best model (i.e. the one with the lowest *NRMSE* score on the test set) can be selected for each RF performance and for the different scenarios. Results are summarized in Table 3.4, which reports for each model the number of selected IMs together with the *NRMSE* scores computed on train and test sets, for both model types.

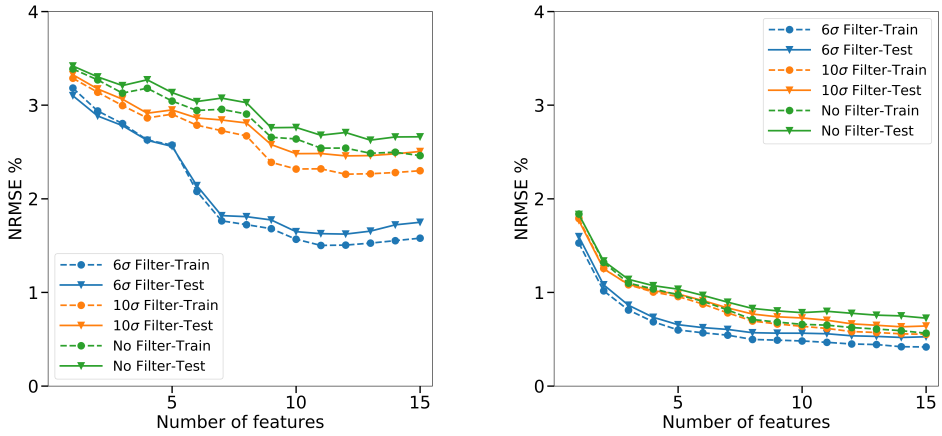


(a) Tx-EVM

(b) Tx-gain

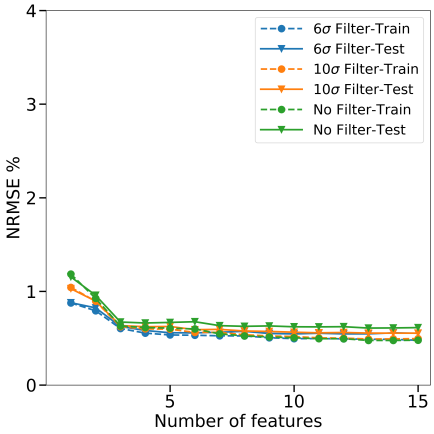
(c) Rx-gain

Figure 3.7: NRMSE score achieved on train and test sets for the different scenarios of learning population - MARS model



(a) Tx-EVM

(b) Tx-gain



(c) Rx-gain

Figure 3.8: NRMSE score achieved on train and test sets for the different scenarios of learning population - SVM model

Table 3.4: Summary of the best results achieved under different scenarios of learning population for the three RF performances

		MARS-NRMSE			***SVM-NRMSE***		
		#IM	*Train*	*Test*	*#IM*	*Train*	*Test*
Original Learning Set	Tx-EVM	15	2.86%	2.89%	13	2.48%	2.63%
	Tx-gain	13	0.85%	0.87%	15	0.56%	0.76%
	Rx-gain	15	0.60%	0.60%	13	0.48%	0.61%
10σ-filtered Learning Set	Tx-EVM	10	2.75%	2.85%	12	2.26%	2.46%
	Tx-gain	15	0.85%	0.84%	14	0.56%	0.63%
	Rx-gain	14	0.57%	0.58%	15	0.49%	0.55%
6σ-filtered Learning Set	Tx-EVM	12	2.52%	2.52%	12	1.51%	1.62%
	Tx-gain	13	0.63%	0.68%	14	0.42%	0.52%
	Rx-gain	15	0.51%	0.55%	12	0.49%	0.54%

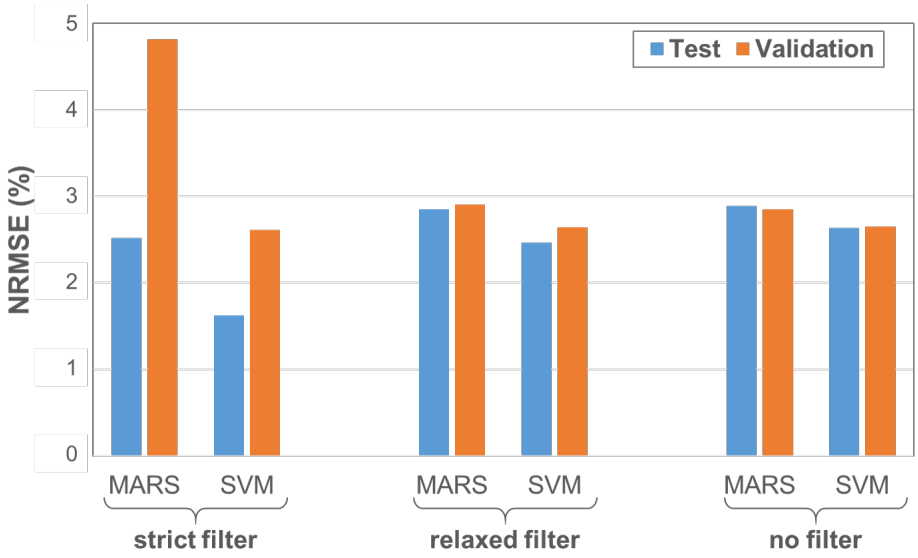
These results confirm the general trends previously observed on the graphs. Indeed, we observe that whatever the learning set, the difference between the *NRMSE* scores computed on training and test sets never exceeds 0.2%, clearly indicating that there is no over-fitting. Regarding the improvement brought by the use of a filter, it can be considered as negligible in case of the relaxed filter with a reduction of the *NRMSE* score that remains inferior to 0.2% over the 3 RF performances for both model types. In case of the strict filter, it is also negligible regarding the Rx-gain and Tx-gain for both model types (*NRMSE* reduction less than 0.2%). It is more significant regarding the Tx-EVM, especially for the SVM model with an (*NRMSE* reduction around 1% while it is only of 0.37% for the MARS model.

Globally, these results are positive for the implementation of the indirect test strategy since they show that it is possible to build quite accurate models for the three RF performances. The best solution is obtained using SVM models constructed on a learning population filtered with a strict filter. In this case, we obtain an accuracy of 0.54% for Rx-gain prediction, 0.52% for Tx-gain prediction and 1.62% for Tx-EVM prediction.

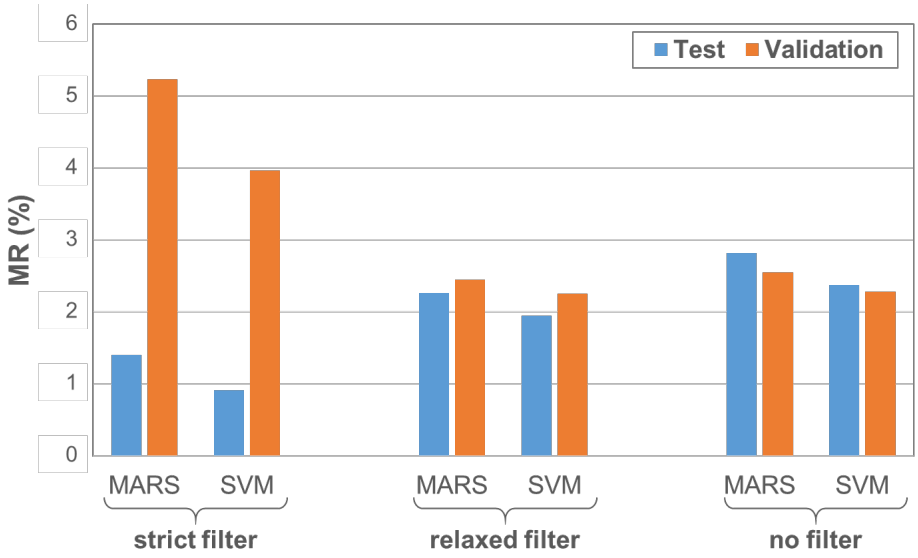
3.5.2 Efficiency of classical indirect test implementation

In this part, we explore the efficiency that can be achieved with a classical indirect test implementation, i.e. all circuits are evaluated using only the indirect test flow and there is no circuit directed to a regular specification test flow (tolerance zone set to zero). Additionally, we also investigate the influence of using (or not) a filter during the initial learning phase onto the fore-mentioned indirect test efficiency. Resulted presented in this section are obtained using the best MARS and SVM models constructed in the training phase for each scenario of learning population. Results are summarized in Figure 3.9, 3.10, and 3.11 for the three RF performances, in which the *NRMSE* and

MR scores achieved on the test and validation sets are compared.

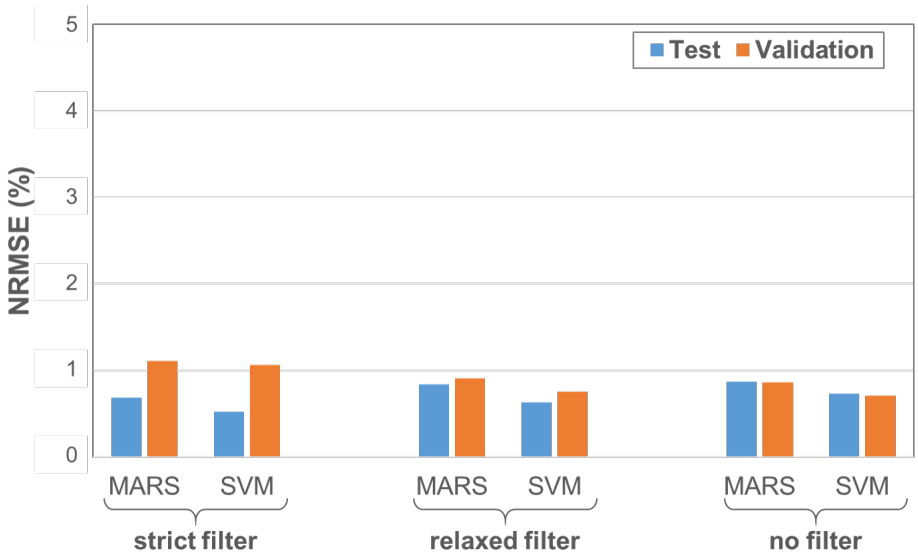


(a) NRMSE

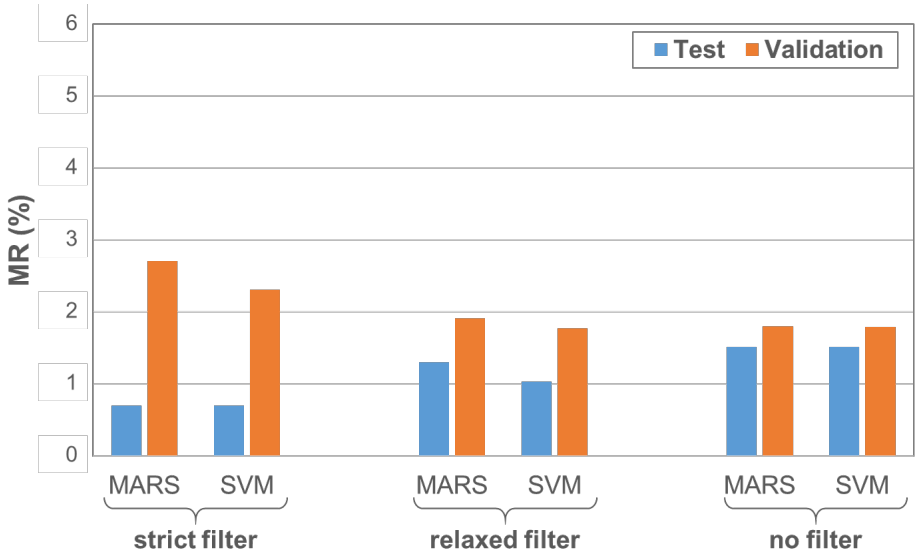


(b) MR

Figure 3.9: NRMSE and MR scores achieved on test and validation sets for the different scenarios of learning population - Tx-EVM

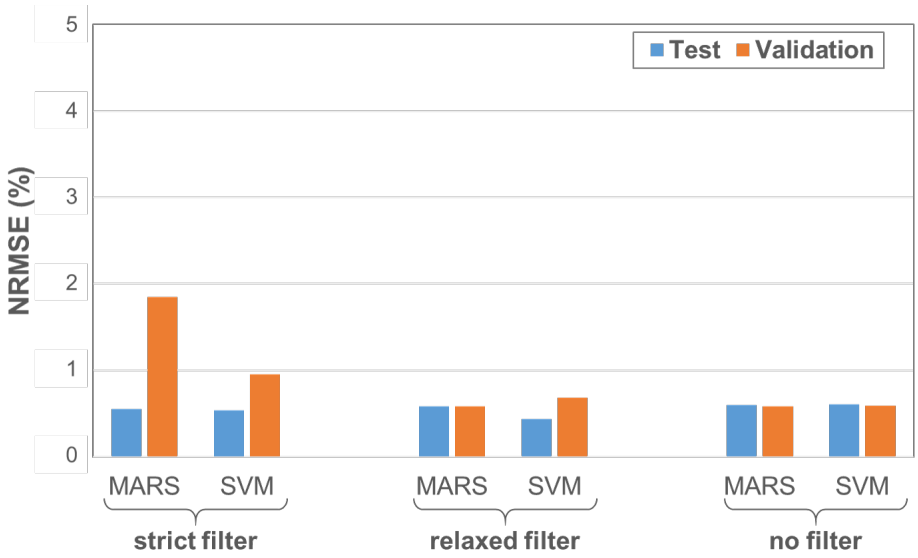


(a) NRMSE

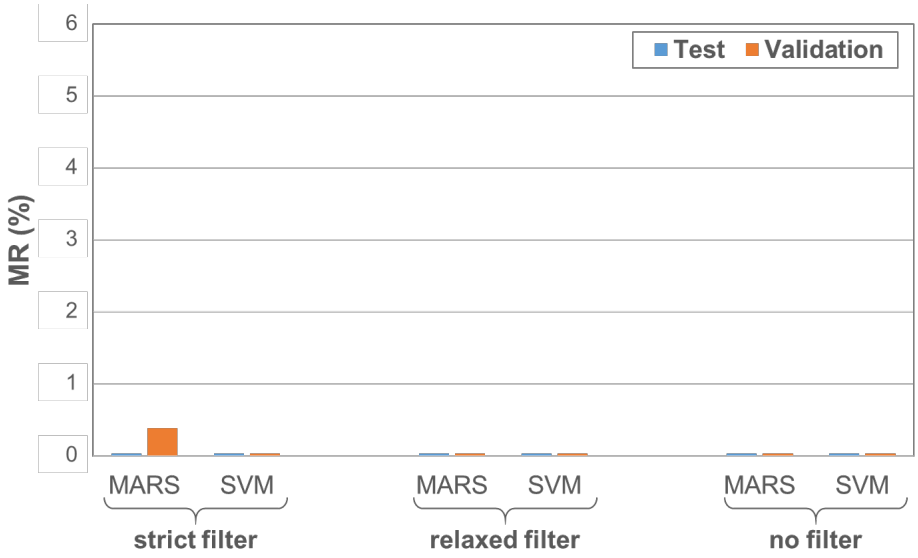


(b) MR

Figure 3.10: NRMSE and MR scores achieved on test and validation sets for the different scenarios of learning population - Tx-gain



(a) NRMSE



(b) MR

Figure 3.11: NRMSE and MR scores achieved on test and validation sets for the different scenarios of learning population - Rx-gain

A first evident comment arises: although working with a filtered population during the initial learning phase permits to improve the quality of the constructed models, these models are nevertheless not able to correctly handle all circuits that can be encountered during the production testing phase, which are emulated by the validation set. Indeed, the best results in terms of *NRMSE* and *MR* scores achieved on the validation set are actually obtained when the models are built on the original learning population. Detailed comments are provided hereafter.

When the strict filter is used, we can observe a significant degradation between the *NRMSE* and *MR* scores determined on the test set and the ones achieved on the validation set. The strongest difference is observed for the Tx-EVM (Figure 3.9). In this case, the *NRMSE* score increases by +2.3% for the MARS and +1% for the SVM model. Similarly, the *MR* score increases by +3.8% for the MARS model and +3.1% for the SVM model. For the Tx-gain (Figure 3.10), we observe a smaller increase of the *NRMSE* score by +0.4% for the MARS model and +0.5% for the SVM model, but still a significant increase of the *MR* score by +2.0% for the MARS model and +1.6% for the SVM model. Finally, for the Rx-gain (Figure 3.11), the impact is mostly visible in case of the MARS model. Indeed, the *NRMSE* score increases by +1.3% and although a perfect *MR* score of 0% is expected from the result on the test set, 0.39% of the circuits of the validation set are misclassified. The degradation of the *NRMSE* score is more limited in case of the SVM model with an increase of only+0.4% and the perfect *MR* score of 0% is preserved. Globally over the three RF performances, SVM models built under this learning scenario outperform MARS models in terms of both *NRMSE* and *MR* scores achieved on the validation set.

When the relaxed filter is used, the degradation of the *NRMSE* and *MR* scores between the test and validation sets lessens. Indeed in this case, the increase of the *NRMSE* and *MR* scores respectively does not exceed +0.3% and +0.7% over the three RF performance, for both model types. Still, SVM models perform slightly better than MARS models.

Finally when the learning is realized on the original population, the difference between the *NRMSE* and *MR* scores determined on the test set and the ones achieved on the validation set becomes negligible. Indeed, the maximum difference observed over the three RF performances is only of 0.06% in the *NRMSE* score and 0.3% in the *MR* score, for both model types (note that the difference in the scores between test and validation sets is positive is some cases and negative in other cases). SVM models present a slight advantage compared to MARS models, but not really significant.

All these results indicate that it is not pertinent to work with a filtered population, since it can entail a strong discrepancy between the test quality estimated during the learning phase and the one encountered during the production phase. Moreover, this experiment shows that the best results achieved on the validation set are obtained when the learning is done on the original learning set. Hence, it is recommended to include circuits with possible extreme values with regards to the indirect measurements in the learning population. This gives some assurance that the accuracy of the models evaluated during the training phase is representative of the one that will be achieved during the production testing phase.

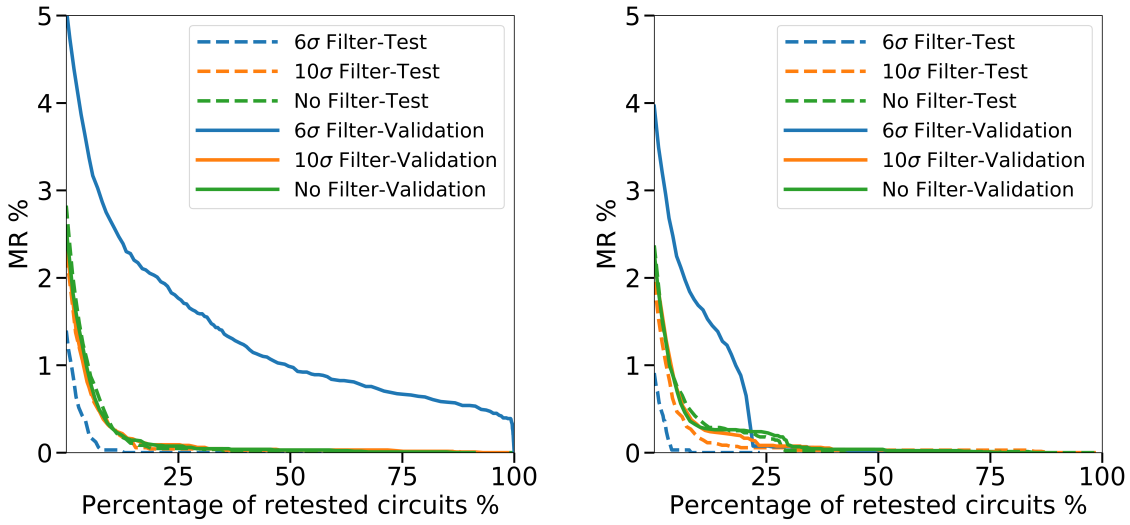
To conclude on this study regarding the classical implementation of the indirect test strategy, a fairly good efficiency is attained for the practical case study investigated in this chapter when the models are built on the original learning set (unfiltered). Indeed, evaluation on the validation set leads to a low *MR* around 2.5% and 1.8% for the Tx-EVM and Tx-gain respectively, and the ideal *MR* of 0% for the Rx-gain for both types of regression models. Globally over the three RF performances, the misclassification rate is around 4%. Note that this misclassification rate is higher than the one achieved on each individual RF performance because the circuits misclassified for a given performance are not necessarily the same than the ones misclassified for another performance.

Despite the drastic testing cost reduction offered by the classical indirect test solution where all the circuits are evaluated based only on low-cost indirect measurements, a misclassification rate around 4% might not be sufficient to comply with the industrial test quality constraints. This motivates the need of investigating on an adaptive two-tier test approach and examining its advantages upon a classical indirect test.In particular, it is possible to attain a very low *MR* score below few tenths of percent with a majority of devices that are tested using only the low-cost indirect measurements.

3.5.3 Efficiency of two-tier adaptive test flow

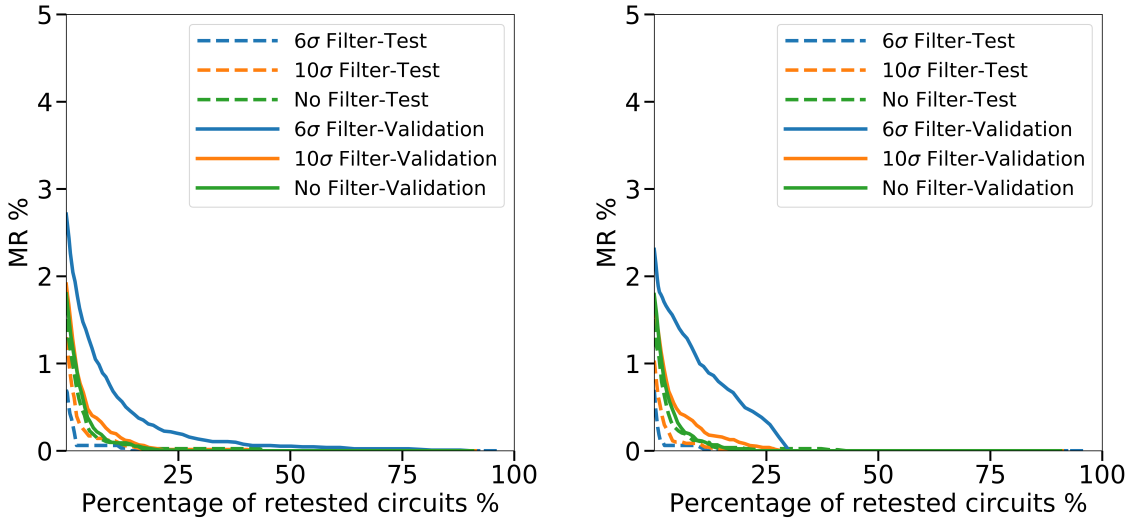
In this section, we present results that show the benefit that can be brought by the implementation of a two-tier adaptive test flow, in particular regarding the trade-off between test quality and test cost. As mentioned in Section 3.2.3, this trade-off depends on the size of the tolerance zone around the test limits.

Results are summarized in Figure 3.12 and 3.13, which report the trade-off curves between *MR* score and percentage of retested circuits obtained by varying the size of the tolerance zone, for both model types. Note that these curves are presented only for the Tx-EVM and Tx-gain performances since the ideal *MR* of 0% is achieved for the Rx-gain without the need of retesting any devices. These results indicate again that the use of a filter during the learning phase (especially the strict one) is not recommended, since there is a huge difference between the trade-off curve evaluated on the test set and the one observed on the validation set. Moreover, the decrease in the *MR* score observed on the validation set is much slower than the one obtained when the learning is performed on the original population. These results also clearly demonstrate that it is possible to significantly improve the test quality compared to a classical indirect test implementation. Indeed, with a learning performed on the original population, there is a rapid decrease of the *MR* score observed on the validation set, which means that the test quality improvement can be obtained with only a limited number of devices that need to be retested through a conventional specification test.



(a) MARS (b) SVM

Figure 3.12: Trade-off curves between MR and percentage of retested devices - Tx-EVM



(a) MARS (b) SVM

Figure 3.13: Trade-off curves between MR and percentage of retested devices - Tx-gain

For the sake of a concrete illustration, an arbitrary target of a *MR* score below 0.1% for each RF performance has been fixed. Based on devices of the test set, the size of the tolerance zone necessary to fulfill this constraint has been determined for each RF

performance; the efficiency of the two-tier adaptive test flow has then been evaluated on the validation set. Results are summarized in Table 3.5 (results obtained under the best learning scenario with no filter applied on the learning population).

Table 3.5: Summary of results achieved by the two-tier adaptive test flow with an *MR* target of 0.1% for each RF performances

	MARS			**SVM**		
	MR (%)		**% retested**	**MR (%)**		**% retested**
	Test	Validation	Validation	Test	Validation	Validation
Tx-EVM	0.10	0.14	15.7	0.10	0.12	29.0
Tx-gain	0.10	0.09	12.0	0.10	0.08	12.4
Rx-gain	0	0	0	0	0	0
ALL RF Perf.	**0.19**	**0.23**	**23.8**	**0.19**	**0.20**	**31.4**

These results confirm that the two-tier adaptive test flow permits to reach a substantial reduction of the test cost while preserving a very good test quality. Indeed, the targeted MR score of 0.1% can be attained for each RF performance; the difference between the MR score anticipated on the test set and the one evaluated on the validation set remains inferior to 0.04%, for both model types. Only a limited number of devices need to be retested to ensure this quality, especially when considering a MARS model, i.e. around 16% for the Tx-EVM, 12% for the Tx-gain, and 0% for the Rx-gain. The percentage of retested devices is significantly higher for the Tx-EVM in case of a SVM model, with a value reaching 29%, while it remains around 12% for the Tx-gain, and 0% for the Rx-gain.

For both model types, the global misclassification rate achieved over the three RF performances is around 0.2%, so higher than the targeted one on each individual RF performance. The global MR score achieved over the three RF performances actually corresponds to the sum of the MR score achieved on each individual performance, indicating that circuits misclassified with respect to a given performance are different than the circuits misclassified for another performance. Regarding the percentage of retested devices, the global percentage over the three RF performances is also higher than the individual percentage on one performance, but inferior to the sum of the individual percentages. This indicates that among all circuits directed to the second tier of the test flow, a number of them present a low confidence for more than one RF performance.

To conclude on this study regarding the implementation of a two-tier adaptive indirect test flow, a very good test quality can be achieved for this practical case study, with only about 0.2% misclassified devices over the three RF performances while a majority of devices are processed using only the low-cost indirect measurements lead-

ing to substantial saving in the test costs. For this adaptive test flow, MARS models seems more performing than SVM models since, for the same level of misclassification rate, more than 76% of the devices are evaluated by the indirect test tier when using MARS models, and only 69% when using SVM models. Finally, note that all these numbers correspond to worst-case results because they are established on a population fabricated with corner process conditions. We can expect lower numbers, especially the percentage of circuits that need to be retested, in the regular context of production testing where circuits are manufactured under normal process conditions.

3.6 Conclusion

In this chapter, we have investigated on a practical case study whether it is possible to benefit of the potential test cost reduction offered by the indirect test strategy without compromising the test quality. We have proposed an original implementation of a two-tier adaptive test flow that relies on the use of a tolerance zone around test limits in order to establish the confidence in the decision proposed by the indirect test; only devices with sufficient confidence are processed by the indirect test while others are directed to a second tier where they are evaluated by a standard specification test. A methodology has been defined in order to make the pertinent choices for the efficient implementation of this test flow.

Particular attention has been paid on the composition of the learning set, especially with regard to the presence of circuits that present outlying values. These circuits can be easily identified with a simple one-dimensional filter. In this study, we observed that it is not pertinent to exclude these circuits from the learning set. Indeed, although working with a filtered population improves the accuracy of the models built in the training phase, it results in a degradation of the test efficiency observed on the validation set, which is representative of the test efficiency that will be achieved during the production testing phase. Nevertheless, it should be highlighted that for this particular study, the circuits exhibit outlying values only with respect to the indirect measurements. The use of a filter might be relevant in the case of circuits that exhibit outlying values with respect to the RF performances.

Finally, results clearly demonstrated the value of the two-tier adaptive test flow, which allows to attain a very good test quality, while achieving a substantial reduction in the test costs. Indeed, in this case study, the misclassification rate attained by a classical implementation of the indirect test strategy remains above few percent, in the best conditions. Using the two-tier indirect test flow, a misclassification rate below few tenths of percent can be achieved with less than 25% of the devices that have to go through a standard specification test. Using the proposed methodology, test engineers have multiple choices at their disposal to ensure an efficient implementation of indirect testing.

Chapter 4

Embedded Indirect Test for Performance Monitoring

4.1 Introduction

In the previous chapters, the concept of an indirect test strategy has been introduced and studied in the context of replacing the classical specification-based testing for analog and RF integrated circuits. The objective is to verify the quality of manufactured devices at the time of their production. However, once the device leaves the production facility and is deployed in the field, issues regarding reliability become crucial, especially with regards to a possible performance degradation due to aging effects. In this chapter, we introduce a new perspective of exploitation of the indirect test strategy for performance monitoring of the device within the application.

Reliability issues have been extensively studied for digital devices and a number of on-line monitoring solutions have been proposed. In contrast, research is more scarce for analog/RF circuits. Authors in [47] investigate the design of an adaptive checker for concurrent error detection based on common mode signal analysis. Authors in [48] proposed a real time estimation to monitor a performance accurately by capturing the distortion performance variation. The use of an embedded temperature sensor to monitor the performance for a RF circuit has been proposed in [49]. A current-based monitor circuit has been proposed in [50] to monitor performance degradation. On the other hand, the use of an indirect test strategy has been limited to implementing a built-in self test (BIST) for analog/RF integrated circuits [18, 51–54], or to perform a post-manufacturing calibration [55, 56]. Hence, to the best of our best knowledge, the use of an indirect test strategy has never been proposed to monitor a performance online.

In this chapter, we delve into how to adapt the indirect test strategy for online performance monitoring of a circuit during its lifetime. The chapter is organized as follows. Section 4.2 introduces the principle and discusses the necessary adaptations. The case study used for the development of a proof-of-concept is then presented in

Section 4.3. Finally, Section 4.4 and Section 4.5 are respectively dedicated to the elaboration of the regression model and the implementation of embedded performance prediction.

4.2 Adaptation of the Indirect Test Strategy for Performance Monitoring

The principle of the proposed strategy for on-line performance monitoring is illustrated in Figure 4.1. As in the classical indirect test implementation, it involves a preliminary learning phase in which the mapping between a given circuit performance and some indirect measurements is established through the construction of a regression model. The main difference is that the learning set should include not only devices affected by process variations but also devices representative of the main wear-out failure mechanisms susceptible to occur during the circuit life. It is therefore preconized that the learning set includes devices that have been submitted to accelerate life tests or burn-in. Once the learning phase is finished, the second phase of the strategy can start. In this phase, every new device deployed in the field predicts its own performance based on model established during the learning phase.

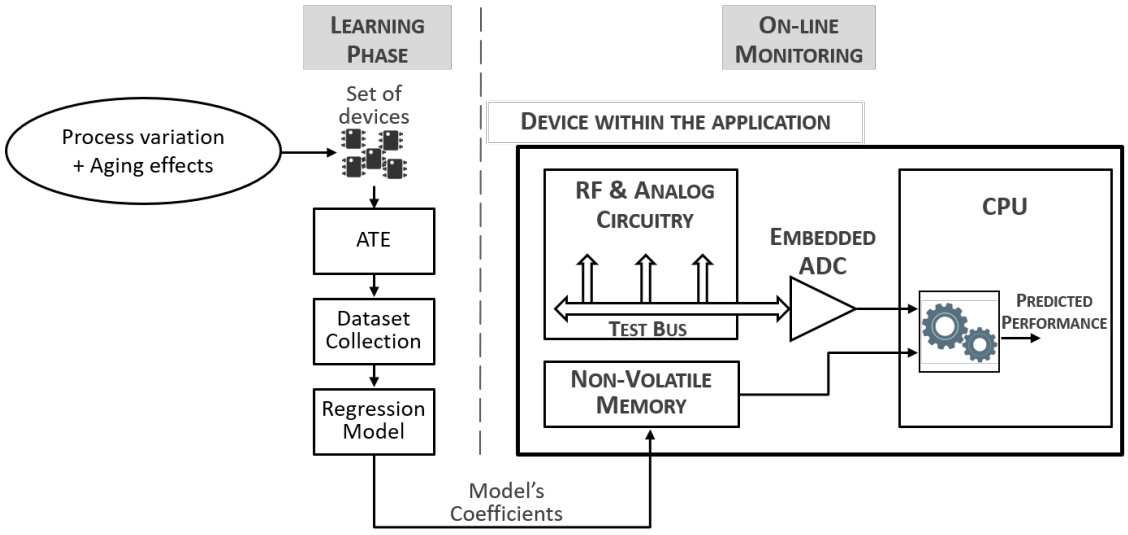


Figure 4.1: On-line performance monitoring based on the indirect test strategy

To perform the embedded prediction, a number of resources are obviously necessary: (i) a dedicated infrastructure (e.g. test bus) in order to access to the internal nodes or structures involved in the indirect measurements, (ii) digitization resources to convert the measured analog values into the digital domain, (iii) a non-volatile memory to store and fetch the coefficients of the established regression model and finally, (iv) a processing

unit to perform the computation of performance prediction. The main requirements on these resources are discussed hereafter. Note that all these resources do not necessarily have to be embedded within the circuit itself, but some of them might be available in the application.

4.2.1 Indirect Measurements

Assembling a comprehensive set of pertinent indirect measurements is a keystone to achieve a successful implementation of an indirect test strategy for production testing. Thus, the same is of course true for an on-line performance monitoring solution based on the indirect test strategy. However, not all types of indirect measurements considered as potential candidates in the context of production testing (cf. Section 1.5) are applicable in the context of an on-line monitoring solution. For example, the standard DC tests usually applied on external pins are realized with the help of the ATE resources and can not be easily performed by the device itself. In the same way, the possibility of changing the test conditions, e.g. the power supply voltage, is not an issue during production testing but it is much more tricky to implement when the device is deployed in the field. Hence, candidates in the context of on-line performance monitoring are embedded measurements that can be realized with a simple test infrastructure (e.g. internal nodes or with built-in sensors accessible with a DC test bus).

Furthermore, the main motivation behind the on-line performance monitoring strategy is to observe and detect any performance deterioration induced by aging effects. The most important IC aging phenomena observed today in nanometric technologies are hot carrier injection (HCI), time dependent dielectric breakdown (TDDB), bias temperature instability (BTI) and electromigration (EM) [57]. These aging phenomena affect not only the digital parts but also the analog/RF parts [58]. They result in internal variations and alterations of the integrated circuit characteristics, such as a shift of the threshold voltage, which can heavily impact the value of the bias voltage of an amplifier, or a shorter gate-oxide breakdown lifetime among other effects [59]. Hence, in order to monitor any performance degradation in-field, it is not sufficient for the indirect measurements to be sensitive to process variations, they also must be sensitive to the main wear-out mechanisms due to aging.

4.2.2 Digitization

Once the indirect measurements and the test infrastructure allowing to access these measurements are defined, the following step is the choice of the digitization resources. Indeed, the measured analog values must be converted into digital values in order to be further processed for the computation of the embedded prediction. It is essential that the quantization error introduced by this conversion does not significantly affect the accuracy of the computed prediction. The choice and the design of the digitization resources in terms of number of bits and measurement range is therefore an important

aspect.

The choice should take into consideration the characteristics of the different indirect measurements used for performance prediction. Indeed, it is likely that the indirect measurements cover a large voltage range while the variation range of each indirect parameter might be small. The digitization resources have to cope with this diversity without comprising the conversion accuracy. In particular, the voltage resolution of the ADC used to perform the conversion of a given indirect parameter must be much smaller than the variation range of this parameter.

The voltage resolution of an ADC is equal to its measurement range (or full scale range), divided by the number of quantization levels, i.e. 2^n , where n is the number of bits of the converter. To ensure an appropriate voltage resolution, it is therefore possible to play either on the measurement range or on the number of bits. In this context, several options can be considered for the design of the required digitization resources, i.e. (i) a single high-resolution ADC with a large measurement range that covers the complete variation range of all indirect measurements, (ii) a single medium-resolution ADC with a programmable measurement range that can be adapted to groups of indirect measurements with a similar order of magnitude in the variation range, or (iii) several low-resolution ADCs, each one with a fixed measurement range perfectly adapted to the variation range of one indirect. The retained solution obviously strongly depends on the case study and will be a compromise between the required silicon area and the conversion accuracy (that impacts the prediction accuracy).

4.2.3 Memory and Arithmetic

A regression model is defined by (i) the structure of the function that relates the indirect measurements to the predicted performance and, (ii) a set of coefficient values that parametrizes the regression function. In order to implement an embedded prediction, it is therefore necessary to have memory as well as arithmetic resources. The memory resources are used to store the value of the coefficients established during the learning phase. These coefficient values obviously need to be permanently stored in the circuit or the system, which implies the use of a non-volatile memory. Alongside the memory, an arithmetic unit must be included in the circuit or system to perform the calculations defined by the established regression model.

It is important to mention that performance monitoring of a device it the field is an auxiliary option to improve the reliability of the system but is not the main core of the application. Therefore, the additional circuitry required to implement the embedded prediction must be minimized. More important, the processing time must be minimized in order to maintain the normal operation of the system without disruption. Hence, it is essential to reduce the number of required operations; one way to achieve this is to use simple prediction models.

Regression model choice: Enriched MLR model

In Chapters 2 and 3 we have seen that usually, when implementing an indirect test strategy in the context of production testing, we tend to use a non-linear regression model (MARS, SVM) or a more complex model such as ensemble methods (Stacking, Random Forest...) to predict the device performance with a high accuracy. However, in the context of on-line performance prediction, the use of these types of models is problematic. Indeed, they involve the storage of a substantial number of coefficients as well as the computation of specific non-linear functions that cannot be straightly implemented with a standard arithmetic unit. Embedded performance computation based of such models is therefore consuming both in terms of memory resources and processing time, which is a strong drawback. Moreover, they usually require a large amount of learning data in order to avoid over-fitting.

An alternative way is to lean on less accurate but easier to implement models, such as linear prediction models. Indeed, as seen in Section 1.4, MLR models are simple models that involves only basic arithmetic operations and that can be easily trained using only small dataset. However, the simplicity of implementation comes at the cost of lower model accuracy.

Nevertheless, the authors in [60] have suggested a strategy to enhance the performance of linear prediction models by creating an enriched set of feature candidates from the initial available feature set. More precisely, they have proposed a Python library to generate non-linear features ($log(x), \sqrt{x}, \frac{1}{x}, x^2, x^3, |x|, e^x$) and combine pairs of features with various operators ($+, -, *$); feature selection is then applied on this enriched set to build linear prediction models. Such an approach seems highly promising in the context of embedded performance prediction, since it has the potential be implemented with few memory resources and only basic arithmetic operations, implying low processing time.

4.3 Case Study

4.3.1 Test vehicle: RF transceiver (NXP JN518x)

The test vehicle under investigation is a wireless microcontroller based on an ultra-low power Arm Cortex-M4 processing core that can operate at a maximum frequency of 48MHz. The device supports both Zigbee 3.0 and thread networking stacks to target and enable the development of Home Automation, Smart Lighting and wireless sensor network applications. To support the different networking stacks, the device includes a 2.4GHz IEEE 802.15.4 compliant transceiver along with a combination of analog and digital peripherals as presented in Figure 4.2, such as an eight channel 12-bit ADC and a 320kB embedded Flash memory.

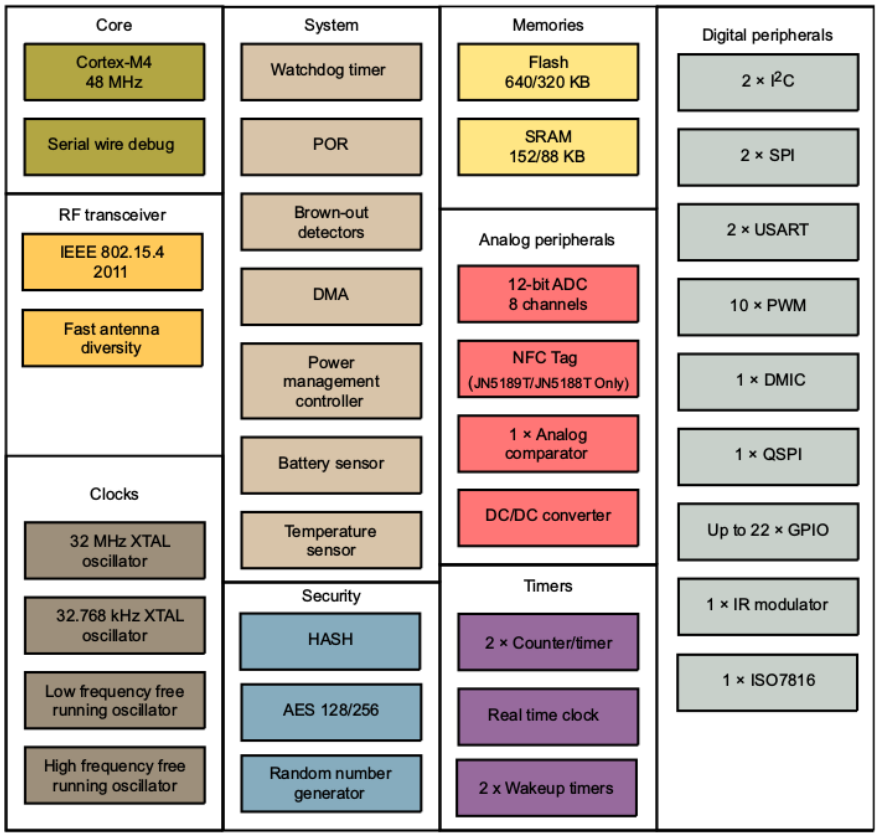


Figure 4.2: High level hardware block diagram of the test vehicle

Alongside with the processing capabilities, the ADC and the embedded Flash memory presented in Figure 4.2, the device also includes an internal analog test structure (DC probes) that connects internal nodes, situated in the different essential blocks, to two General Purpose Input Output (GPIO) pins. Moreover, the internal nodes are also connected to the 12-bit ADC, which permits the main core to process and monitor those internal measurements. Based on the presented and highlighted capabilities of the test vehicle, we believe that the device under study possesses the needed requirements to implement an on-line performance monitoring based on an indirect test strategy.

4.3.2 Dataset Collection: Measurement campaign

The target in this case study is to monitor on-line the transmitted power level of the in-field device based on the indirect test strategy, thus a regression model is needed to predict the power level with the help of the available indirect measurements. Of course building a regression model necessitates the collection of sufficient learning data in which an adequate variation on the level of the transmitted power is observed. Nonetheless, since we are implementing such a strategy on a newly mass-produced device, it is quite impossible to obtain the value of the transmitted power level and the different indirect

measurements data impacted with aging effects; keeping in mind that the implementation of an on-line performance monitor has not been considered during the design phase of the device.

Nonetheless, by changing the configuration of two internal registers that directly impact the transmitted power level, we are able to emulate a power level deterioration within the device in order to imitate the effect of aging and thus obtain an adequate power level variation. Each register is configured using four bits (16 configurations), thus re-configuring both registers would result in 256 different possibilities to operate the transmission block of the device, one of which corresponds to the nominal configuration of the device. Exploiting the capability of re-configuring the device simplifies the task of collecting real and representative data that could be used to build a regression model.

In this experiment, we collected test data from four different JN518X integrated circuits on the V93K industrial test platform from Advantest as illustrated in Figure 4.3. For each circuit we iterate through the 256 possible configurations while measuring the transmitted power level that we wish to predict along with the eleven available test resources (indirect measurements) within the transmitter block (DC30 to DC40) resulting in 1024 observations and 12,288 (1024 instances x 12 measurements) values in total. Certainly the collected dataset is considered as small when compared with the other datasets we have used in the previous chapters. Nonetheless, in this case study we are studying the feasibility of achieving an on-line performance monitor based on the indirect test strategy and thus we are not expecting an industrial accuracy level from the prediction model; the case study is considered as a proof-of-concept rather than being an established method to monitor a performance on-line.

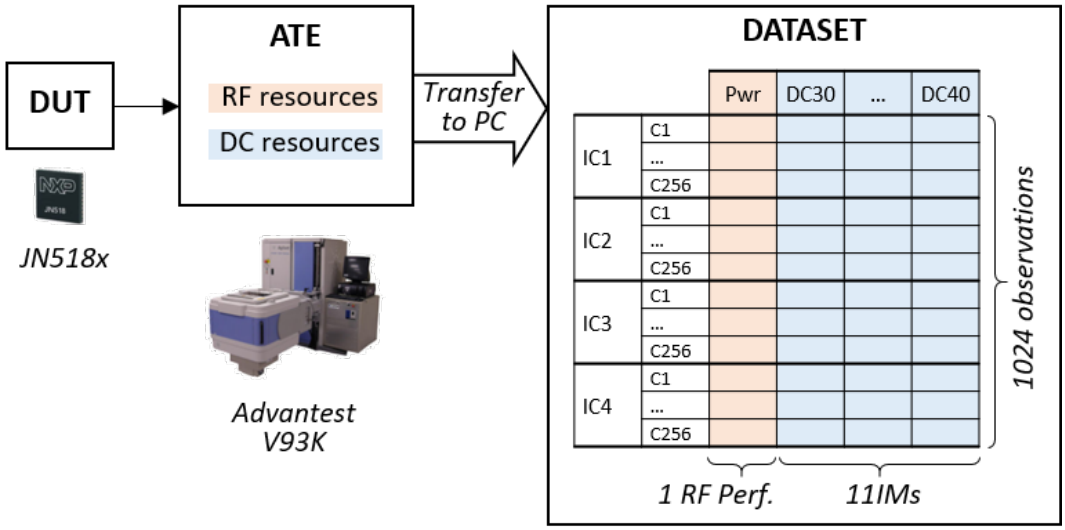


Figure 4.3: Measurement campaign for dataset collection

4.3.3 Dataset Analysis

RF performance: transmitted power level

The histogram of the power level measurements performed with the help of the RF ATE resources in the Advantest V93K across the 1024 observations is presented in Figure 4.4. A clear variation of the power level is observed across the 1024 observations. For this device, the typical power level value under the nominal configuration is $11.4dBm$. The minimal value observed on the dataset is around $8.1dBm$ and the maximal value around $13.2dBm$, which corresponds to a variation of about -29% and $+16\%$ from the typical value. This variation is relatively limited, but significant enough in the objective of building a regression model that is able to predict this performance. However, it is important to confirm that the variation is mainly related to the 256 possible internal register configurations rather than the variability from one IC to another.

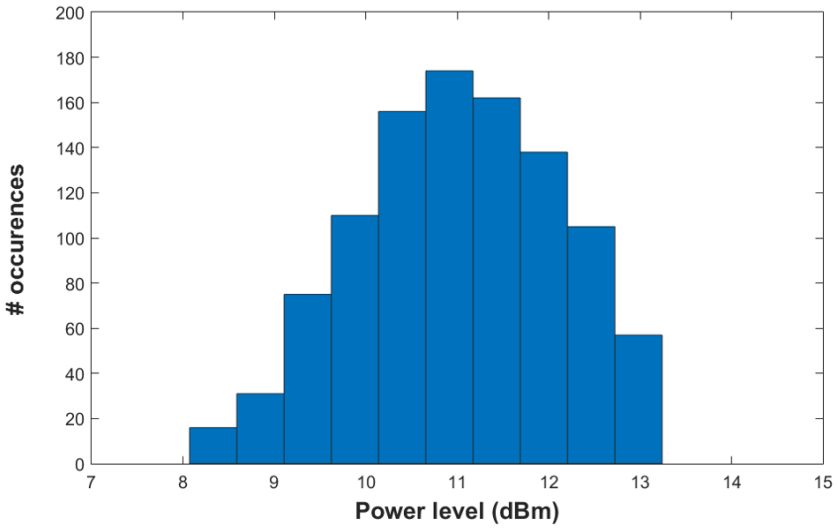


Figure 4.4: Histogram of the transmitted power level measurements for the 1024 observations

In Figure 4.5 we represent the boxplot of the power level measured under 256 configurations for the four different ICs. This figure clearly shows that there is a very low variability between circuits. The difference between circuits does not exceed $0.36dB$ for the mean value, $0.41dB$ for the minimal value, and $0.30dB$ for the maximal variation, which corresponds to a circuit variability of less than 5%.

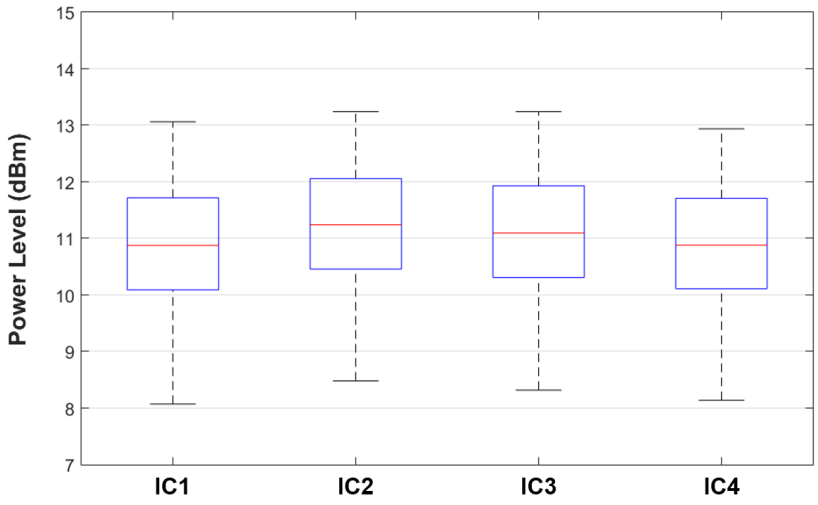


Figure 4.5: Boxplot of the power level measured under the 256 configurations for the four ICs

Indirect measurements

Figure 4.6 presents a boxplot of the DC values measured on the 1024 observations, for the eleven indirect measurements. This figure reveals that the indirect measurements are dispersed over a large range of DC values, spreading from 0 to $1.15V$. However, it is not a uniform distribution and the indirect measurements can be arranged in four different groups, depending on their DC value range:

- DC30, DC31, DC32, DC34, DC36, DC38 and DC40: values between $1.04V$ and $1.15V$
- DC39: values between $0.59V$ and $0.60V$
- DC33: values between $129mV$ and $138mV$
- DC35 and DC37: values between $2.3mV$ and $2.5mV$

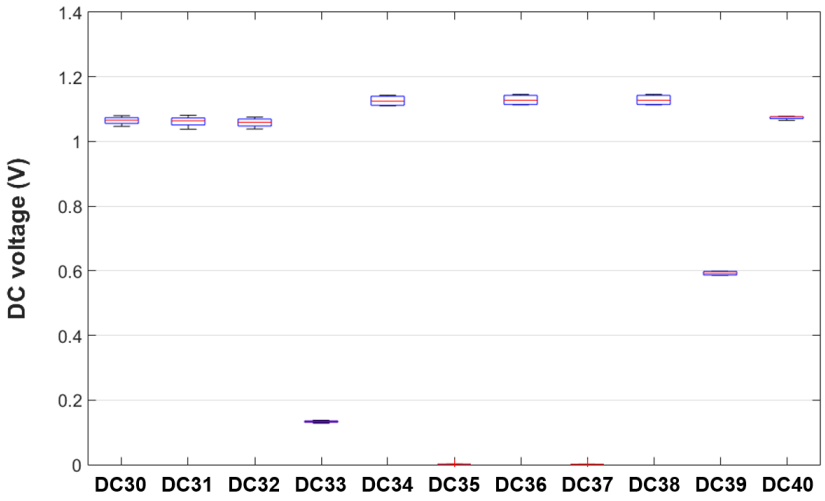


Figure 4.6: Boxplot of the DC values measured on the 1024 observations, for the eleven indirect measurements

This is an important point because, in the perspective of an embedded prediction, it means that the circuit should be equipped with an infrastructure able to perform DC measurements in different ranges. In addition, because of these different ranges, it is not easy to comment and compare the variations observed on each indirect measurement. Therefore, we have computed the relative deviation from the mean (expressed in %) for each indirect measurement.

Figure 4.7 presents a boxplot of this relative deviation for the eleven indirect measurements. This figure reveals that most of the indirect measurements present a small variation over the 1024 observations, i.e. a maximum variation of around 5%. Only two indirect measurements, namely DC35 and DC37, exhibit a significant variation over the 1024 observations, i.e. a maximum variation that exceeds 10%.

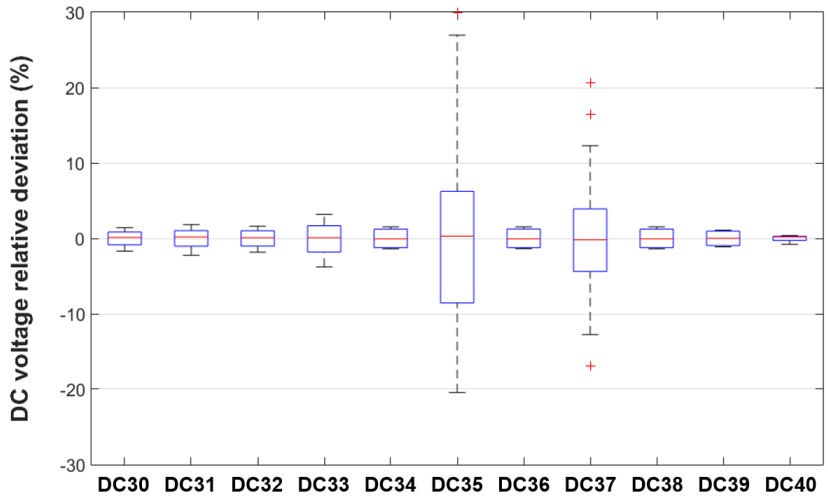


Figure 4.7: Boxplot of the relative DC deviation measured on the 1024 observations, for the eleven indirect measurements

However here again, it is important to establish whether this variation is related to the internal register configurations or to the variability from one integrated circuit to another. Therefore, we have paid attention to the mean relative DC deviation over the four integrated circuits, which is representative of the influence of the circuit configuration, on the one hand, and to the mean relative DC deviation over the 256 configurations, which is representative of the influence of the circuit variability, on the other hand. Results are summarized in Figures 4.8 and 4.9 respectively.

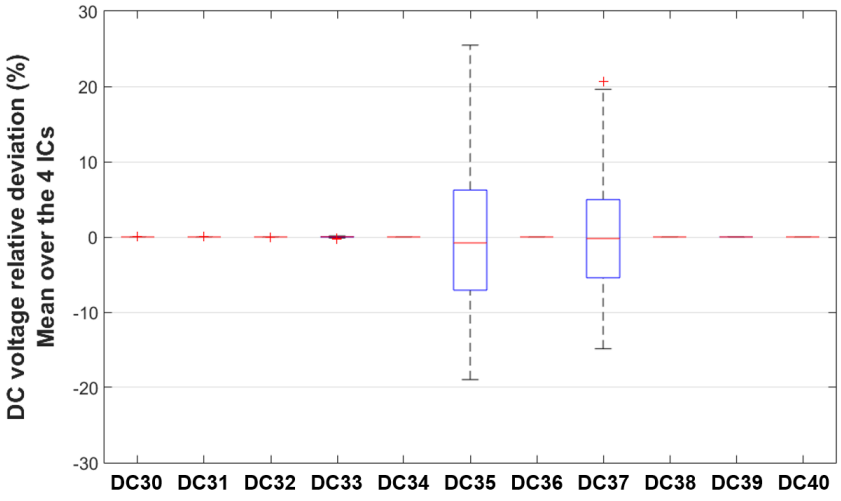


Figure 4.8: Boxplot of the mean relative DC deviation over the four ICs, for the eleven indirect measurements

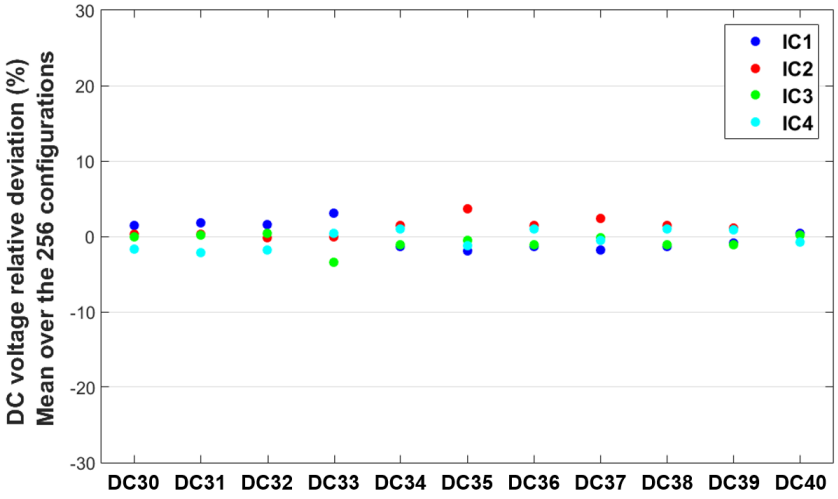


Figure 4.9: Mean relative DC deviation over the 256 configurations, for the eleven indirect measurements

Figure 4.8 clearly shows that only DC35 and DC37 are actually impacted by the circuit configuration with a variation that exceed 20%, all the other indirect measurements presenting an almost constant value (variation lower than 0.03%). In contrast, Figure 4.9 shows that the mean relative DC deviation over the 256 configurations presents a comparable variation range for all indirect measurements, indicating that all the indirect measurements have a similar behavior with respect to the circuit variability. Moreover, the difference from one circuit to another is quite small (maximum difference around 5%). Still, it is important to note that for DC30, DC31, DC32, DC33, DC34, DC36, DC38, DC39, and DC40, this difference is higher than the maximum variation induced by the circuit configuration.

To better illustrate this point, Figures 4.10 and 4.11 respectively present the distribution of DC30 and DC35 measurements under the 256 configurations, for the four different integrated circuits. It can be observed that DC35 exhibits a comparable distribution for the four integrated circuits, with a similar mean value and a similar dispersion. This dispersion is directly related to the circuit configuration. In contrast for DC30, there is no dispersion induced by the circuit configuration, though there is a different DC value for each integrated circuit.

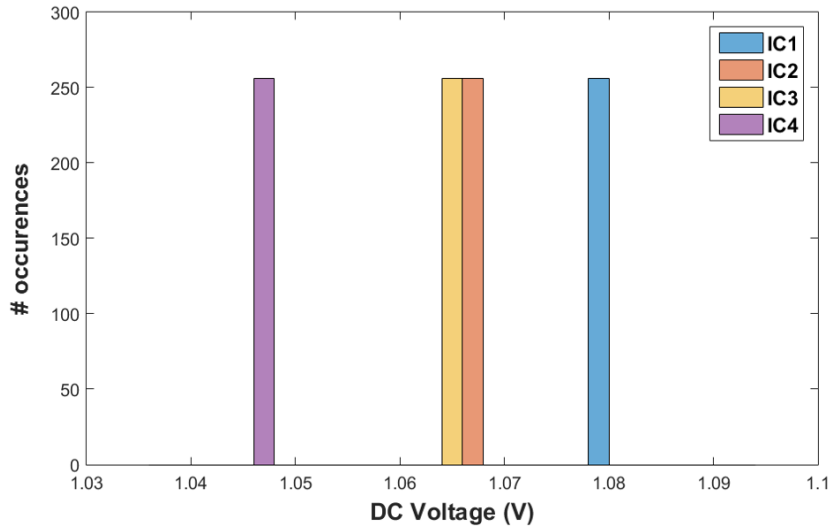


Figure 4.10: Histogram of DC30 measurements under the 256 configurations, for the four ICs

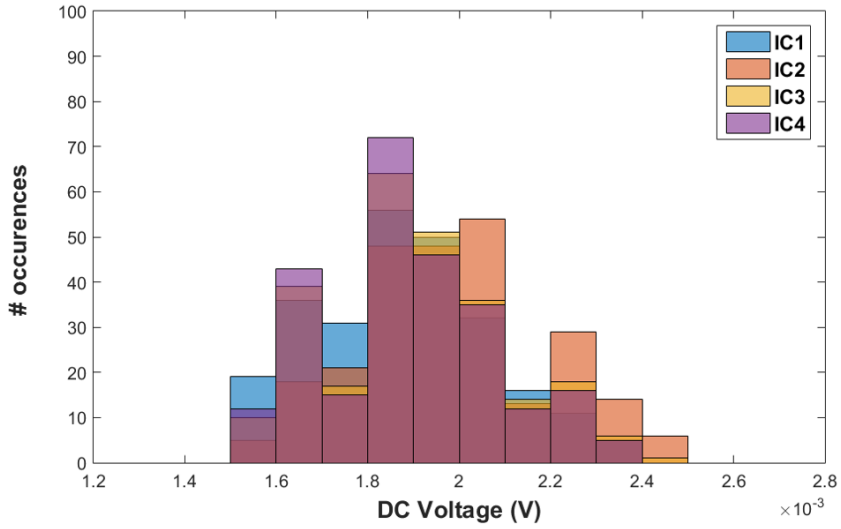


Figure 4.11: Histogram of DC35 measurements under the 256 configurations, for the four ICs

This preliminary analysis reveals that measured values for DC35 and DC37 are mainly determined by the circuit configuration while measured value for all the other indirect measurements mainly come from the manufacturing process. Since the circuit configuration has an influence on the power level along with DC35 and DC37, those two indirect measurements are considered as pertinent and valuable to establish a regression model that is able to predict the power level.

Summary

Let us recapitulate the main points presented in this section. We have seen that it is possible to change the internal register configuration in order to create a sufficient power level variation instead of using a substantial amount of circuits to build a more comprehensive dataset. The four different circuits used in this case study behave similarly with respect to the power level variation created by the different internal register configuration. On the other hand, inconsistent behavior can be observed regarding the variations of the different indirect measurements. Indeed, only two (DC35 and DC37) out of the eleven available indirect measurements exhibit a variation related to the internal register configuration higher than the variability observed due to the circuit variation.

It is clear that this situation is not the best context to implement an efficient solution for an on-line performance monitoring based on an indirect test strategy. However, it might be sufficient to establish a proof-of-concept. The expectation is that, despite the weaknesses of this dataset, we can build a regression model that combines the different indirect measurements with a sufficient accuracy to predict the power level. This regression model, once established, will the be used to perform an embedded performance prediction in addition to the main device application, and therefore to validate the concept of the proposed strategy.

4.4 Model Elaboration

4.4.1 Preliminary study: choice of model type

As commented in Section 4.2.3, a possible solution to improve the accuracy of a standard MLR model is to enrich the space of candidates available for the construction of the model during the training phase by including, not only the original IMs, but also non-linear transformations of the original IMs as well as combinations of pairs of IMs. In our context of an embedded prediction, it is important to ensure that the computation of the non-linear transformations as well as the interaction between IMs can be computed within the circuit.

Regarding the non-linear transformations, some transformations such as $\frac{1}{x}$ and x^2 can be implemented at low-cost because they require only a limited number of elementary arithmetic operations. In contrast, other transformations such as $\log(x)$, $\sqrt{x}$ and $\exp(x)$ would require much longer processing times and they are only approximations. Indeed, their exact computation is not feasible with elementary arithmetic operations; instead, numerical algorithms that involve many elementary arithmetic operations have to be used to compute an approximation. In this study, we consider only transformations that can be implemented at low-cost, i.e. $\frac{1}{x}$ and x^2.

Regarding the interaction between IMs, all combinations of pairs of IMs using the four elementary operators $(+, -, *, /)$ can be easily implemented. However, combinations using $(+)$ and $(-)$ operators are intrinsically present in the model; therefore only combinations using $(*)$ and $(/)$ operators are considered in this study.

Globally, with the considered non-linear transformations and interactions, an enriched space of 209 candidates has been generated from the original space of eleven IM candidates. MLR models have been constructed to predict the power level, applying feature selection while using SFS based on multiple linear regression on both the original space and the enriched one.

Figure 4.12 reports the accuracy of the constructed models in both cases, for a number of features comprised between one and five. This figure shows the improvement brought by the enrichment of the candidate space, with a reduction of the NRMSE score of about 0.025dB, which corresponds to an accuracy improvement of about 6%.

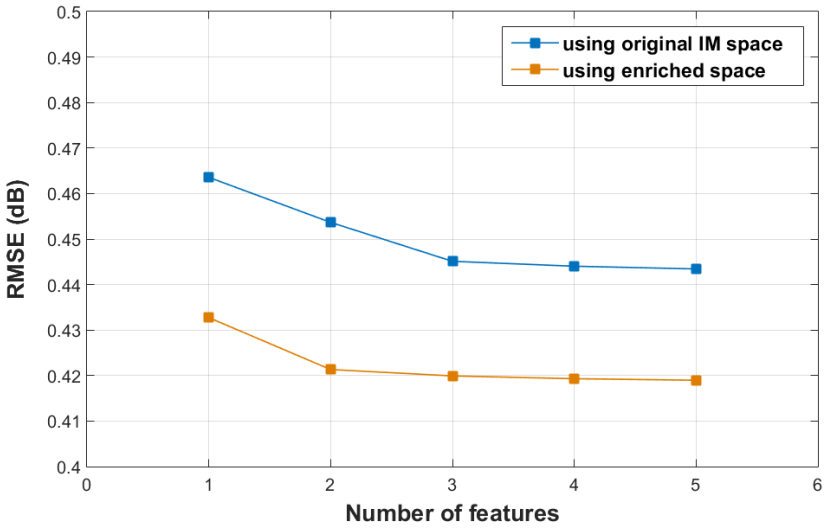


Figure 4.12: Comparison of MLR models constructed on the original IM space or on the enriched one

The retained solution for all following experiments is the use of an MLR model defined by Equation 4.1 that would be used to predict the power level $\hat{P}_i$ and which is constructed on the enriched space with 3 features that involve four IMs:

- Feature 1: $DC39/DC35$
- Feature 2: $1/DC37^2$
- Feature 3: $DC30 * DC35$

$$\hat{P}_i = a_0 + a_1 * (\frac{DC39_i}{DC35_i}) + a_2 * (\frac{1}{DC37_i^2}) + a3 * (DC30_i * DC35_i) \quad (4.1)$$

where a_0, a_1, a_2, a_3 are the model coefficients that must be stored in the available Flash memory to be able to produce a power level prediction within the circuit.

4.4.2 Implementation in the case study

The main objective of this case study is to detect a possible power level degradation of the circuit during its operation and alongside its main application. Thus, the idea is to perform an embedded power level prediction by implementing a regression model. The regression model is established during an initial learning phase using other circuits as presented in Section 4.2, Figure 4.1. The available dataset is comprised of four different circuits for which we have their test data, i.e. the measurements performed on the ATE which are then used to build the regression model.

Even though the available dataset is limited and does not have a substantial amount of data, it is possible to develop an experimental protocol to validate our proposed proof-of-concept. Indeed, it is necessary to evaluate the established regression model on unseen instances. One way to perform such a task is through employing a Leave-one-out cross-validation technique [61]. The main idea is to use all the available instances except one to establish the prediction model and evaluate the model using the one remaining instance. Although, in our case we will consider a circuit as the leave-one-out sample instead of considering one observation out of 1024 available observations. Thus, in reality it is a Leave-n-out cross validation where we perform the learning phase on three circuits (768 observations) and then we predict the power level of the remaining circuit across its 256 different configurations to eventually evaluate the established regression model. As a result, we have four possible learning combinations; each one of them is studied and the results are summarized in Table 4.1 which reports the *RMSE* score achieved for each learning and validation sets.

- Combination 1: Learning: IC1, IC2, IC3 Validation: IC4
- Combination 2: Learning: IC1, IC2, IC4 Validation: IC3
- Combination 3: Learning: IC1, IC3, IC4 Validation: IC2
- Combination 4: Learning: IC2, IC3, IC4 Validation: IC1

Table 4.1: Training and Validation *RMSE* scores for the four possible learning combinations

	Combination 1	**Combination 2**	**Combination 3**	**Combination 4**
***RMSE* score - Training**	0.421dB	0.418dB	0.422dB	0.418dB
***RMSE* score - Validation**	0.417dB	0.427dB	0.415dB	0.426dB

By exploring the results presented in Table 4.1, it is evident that all the different combinations lead to similar results, the different regression models achieve basically the same accuracy on the learning and validation sets. It is also important to highlight that none of the established regression models suffer from over-fitting, given that the difference in accuracy between the learning and validation set is clearly minimal. Therefore, from this point onward, only combination 1 will be considered and all the results presented in the following sections will be based on this combination.

Once the regression model is selected, the ability of the model to detect any performance deterioration, by monitoring the power level prediction, should be evaluated. Hence, the predicted power level variation $\widehat{\Delta P_i}$ has been computed for each configuration with the use of the following Equation 4.2:

$$\widehat{\Delta P_i} = \hat{P}_i - P_{nom} \tag{4.2}$$

where $\hat{P}_i$ is the predicted power level in configuration i and P_{nom} is the power level measured on the ATE for the nominal configuration. Moreover, the prediction error has also been computed with the help of following Equation 4.3, where ΔP_i is the power level variation measured on the ATE in configuration i:

$$\varepsilon_i = \widehat{\Delta P_i} - \Delta P_i \tag{4.3}$$

Results are illustrated in Figures 4.13 and 4.14, which present respectively the predicted power level variation versus the measured one, and the normalized distribution of the prediction error. Numerical results are summarized in Table 4.2, which reports the mean, the standard deviation and the maximum values of the prediction error observed over the 256 configurations.

By examining Figure 4.13, it is clear that the regression model perform acceptably well considering the different shortcomings, i.e. the use of a linear prediction model and the limited size of the dataset. Indeed, when analyzing also the results presented in Figure 4.14 and in Table 4.2, it appears that the prediction error follows a Gaussian distribution and that roughly most of the predicted samples have a prediction error smaller than three times the standard deviation of the prediction error. Moreover, we notice that the spread of the samples around the ideal regression line is uneven across the distribution, where it is evident that the spread of samples in the center is higher than the spread at the extremities.

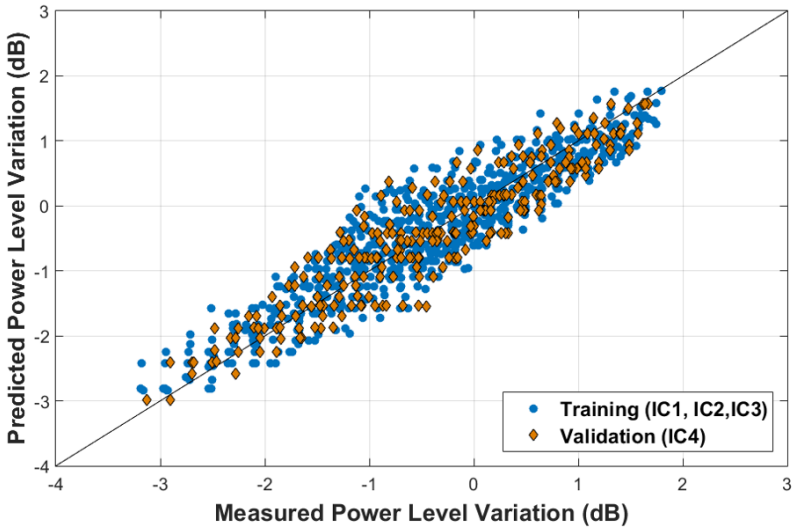


Figure 4.13: Predicted power level variation vs. measured power level variation

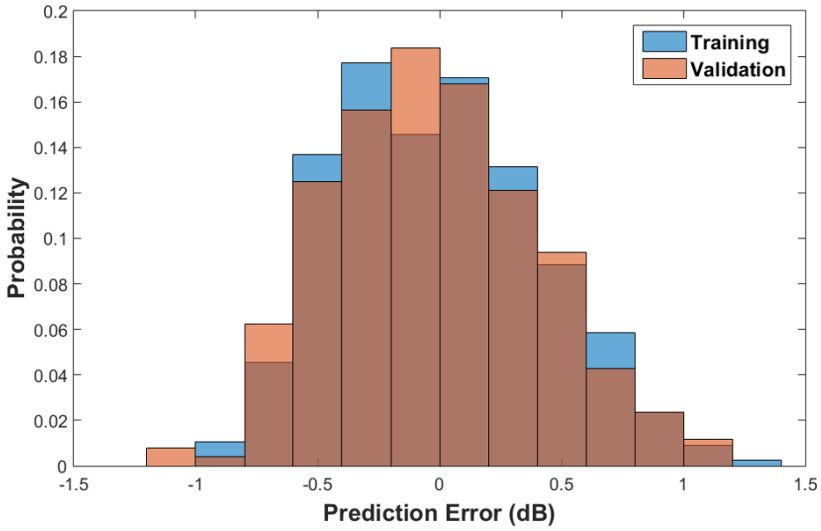


Figure 4.14: Normalized distribution of the prediction error

Table 4.2: Statistics of the prediction error on the 256 configurations

	All configurations		
	Mean (dB)	Standard deviation (dB)	Max (dB)
Prediction error - Training	0.00	0.42	1.30
Prediction error - Validation	-0.02	0.42	1.19

Now, from the presented results, we can (i) determine a limit on the predicted value that will be used during the on-line monitoring process to determine whether a circuit is affected by a performance degradation or not, and (ii) evaluate the efficiency of the on-line monitoring process by computing a priori the detection range of power level degradation. The principle is illustrated in Figure 4.15 and it relies on three steps:

- First, boundaries are determined around the ideal regression line in order to delimit a zone that contains all predicted values. Classically, these boundaries are determined considering 3 times the standard deviation of the prediction error ($\varepsilon = 3\sigma$)
- Then, the intersection between the lower boundary and the x-axis origin (i.e. no performance degradation) is used to determine a limit on the predicted value ($\widehat{\Delta P}|_{lim}$). This limit will be used during the on-line monitoring process to decide whether a circuit is affected by a performance degradation or not. Any circuit with a predicted value below $\widehat{\Delta P}|_{lim}$ will be flagged as a degraded circuit. Note that this limit ensures that all circuits with a predicted value above $\widehat{\Delta P}|_{lim}$ are indeed circuits with a performance equal or superior to the typical specification.
- Finally, the intersection between $\widehat{\Delta P}|_{lim}$ and the higher boundary determines the separation ($\Delta P|_{det}$) between the regions of certain and possible performance degradation detection. All circuits with an actual power level degradation bigger than $\Delta P|_{det}$ will assuredly be detected as degraded circuits by the on-line monitoring process, while for circuits with a power level degradation comprised in the interval [$\Delta P|_{det}$;0], detection might be possible but is not guaranteed. The value of $\Delta P|_{det}$ is therefore an indicator of the on-line monitoring process efficiency; the smaller this value, the better the efficiency.

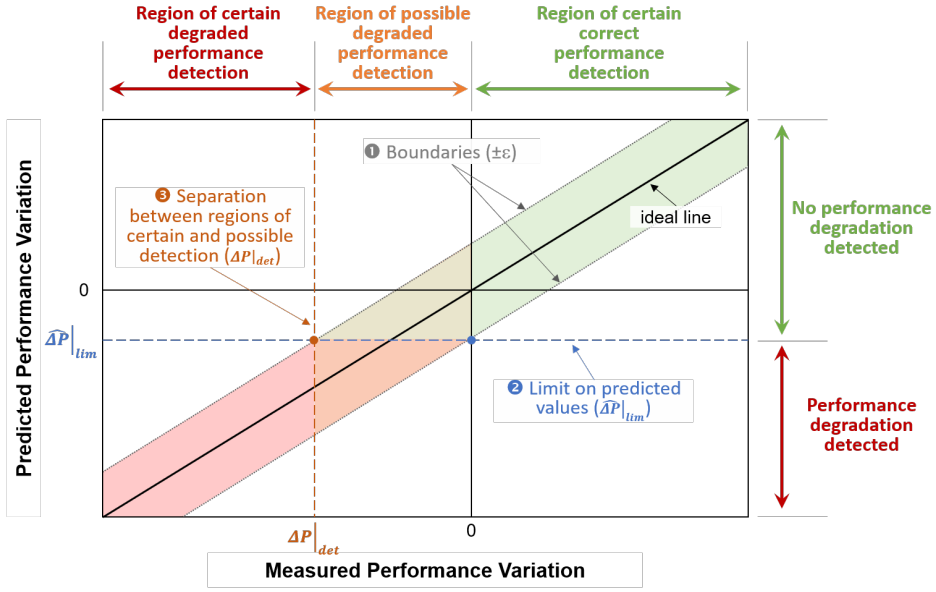


Figure 4.15: Principle of limit value determination and resulting detection range

Note that the efficiency is directly related to the model quality. Indeed, it is clear that the better the model quality, the lower the prediction error, the closer the boundaries from the ideal regression line, and therefore the larger the certain detection range. This information is very important during the elaboration of the on-line performance monitoring process to evaluate whether the achieved detection range is sufficient for the targeted application, or whether further efforts should be deployed to construct a regression model of better quality.

For the case study considered in this chapter, the standard deviation of the prediction error on the training set is $0.42dB$. The limit of the predicted power level variation is therefore placed at $-1.26dB$ and the certain detection range is $[-\infty;$-2.52$]$, as illustrated in Figure 4.16. It can be observed that only few observations are within the region of certain detection. Of course, it is not a favorable situation to finely evaluate the efficiency of the on-line monitoring process. However, we can still exploit these data to present a proof-of-concept.

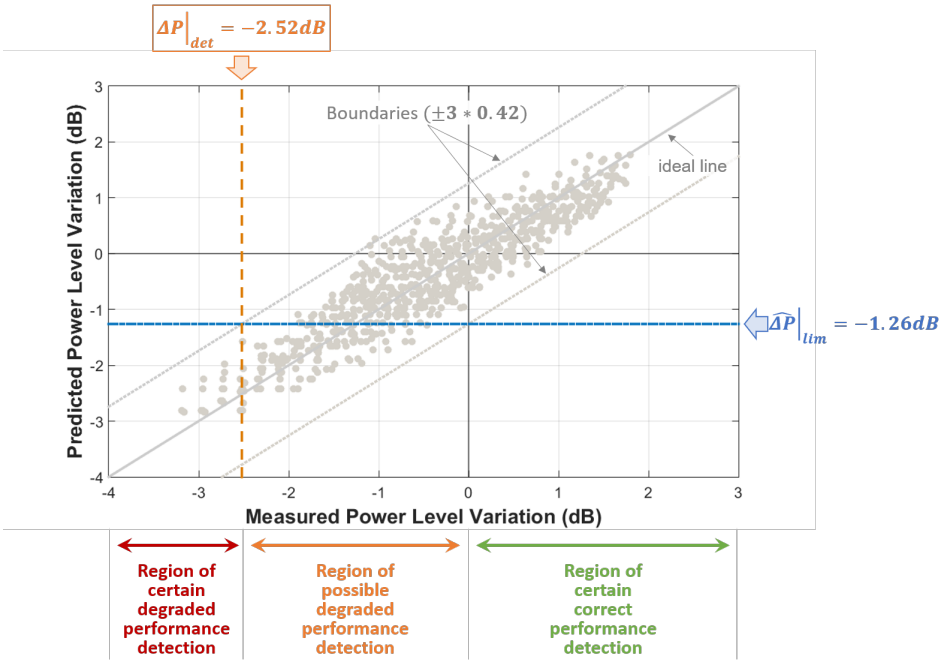


Figure 4.16: Illustration of limit value and detection range for the case study

4.5 Embedded Prediction

In this section, we apply the on-line monitoring process to the case study. More precisely, we use the regression model elaborated in the previous section from ATE measurements on IC1, IC2, and IC3 to perform embedded power level prediction on IC4. We choose to evaluate the proposed solution using two circuit configurations, i.e. C172 and C241. C172 is the nominal circuit configuration; the on-line monitoring process should therefore flag IC4 as a correct circuit in this first case. C241 is a degraded configuration, in which the learning circuits exhibit a power level degradation around $-3.17dB$. This power level degradation is within the region of certain detection; the on-line monitoring process should therefore flag IC4 as a degraded circuit in this second case.

This section firstly presents an exploratory study on the requirements of the circuitry dedicated to embedded measurements. The practical implementation of the proposed solution on the case study is then detailed and experimental results are discussed.

4.5.1 Theoretical study on ADC resolution

As described in Section 4.2, the proposed strategy requires that the product is equipped with a test infrastructure that allows access to the selected indirect measurements as well as digitization of the embedded measurements. In this section, we realize an exploratory study to investigate which are the required performances of the digitizing resources in terms of resolution, assuming that the measurement range of the digitizing resources can be adapted to the variation range of each indirect measurement involved in the prediction model.

Simulation experiments have been conducted considering an ideal ADC model and varying the resolution of this ADC. The measurement range of the ADC has been adapted to the variation range of each indirect measurement, adding a 10% margin on the observed min/max values. Table 4.3 summarizes the ADC measurement range used for each indirect measurement.

Table 4.3: ADC measurement range used for each IM involved in the prediction model

	Observed variation range on training data	**Adapted ADC Measurement range**
DC30	[1.047 V; 1.080 V]	[0.942 V; 1.188 V]
DC35	[1.528 mV; 2.495 mV]	[1.375 mV; 2.744 mV]
DC37	[1.130 mV; 1.642 mV]	[1.017 mV; 1.806 mV]
DC39	[0.586 V; 0.599 V]	[0.527 V; 0.685 V]

Taking into account these measurement ranges, the quantized values of each indirect measurement are first determined for different values of the ADC resolution. The computation of the predicted power level variation is then realized using these quantized indirect measurement values instead of the original indirect measurement values measured by the ATE.

Results are summarized in Figure 4.17, which plots the evolution of the prediction error observed on IC4 (RMSE score evaluated over the 256 circuit configurations) according to the ADC resolution. This figure shows that a constant RMSE score of $0.417dB$ is observed for an ADC resolution from 14 bits down to 6 bits. This RMSE scores is identical to the one computed when the prediction is achieved with the original indirect measurements values measured by the ATE, indicating that the prediction accuracy is fully preserved. An increase of the RMSE score appears when the ADC resolution is below 6 bits. However, it is quite small for 5-bit and 4-bit resolutions, i.e. an increase of only +1.4% for 5-bit resolution and +6% for 4-bit resolution ADCs. the impact on the accuracy of predicted values is therefore limited. The increase of the RMSE score is significantly higher when the ADC resolution falls below 4 bits, i.e. an

increase of +23% for 3-bit resolution, +72% for 2-bit resolution and +189% for 1-bit resolution ADCs. Obviously in these last situations, the accuracy of predicted values is strongly degraded.

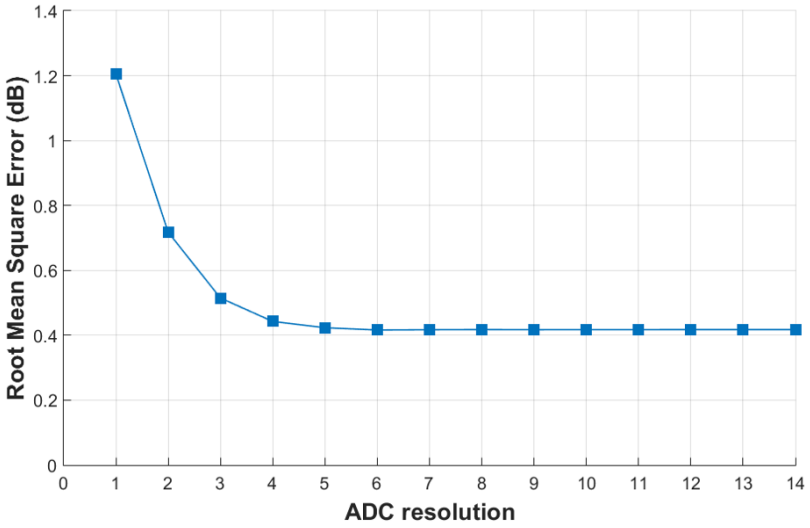


Figure 4.17: Influence of ADC resolution on prediction error on IC4

These results suggest interesting perspectives for the design of the required test infrastructure. Indeed, they show that there is no significant degradation in the prediction accuracy due to the use of digitized embedded measurements, provided that each measurement is digitized with a resolution of at least 4 bits. This means that the use of a high-resolution ADC is not necessarily required. Instead, a low-resolution ADC can be used, but it should have an adaptable measurement range. Alternatively, dedicated circuitry might be included in the test infrastructure in order to apply shift/amplification operations on the indirect measurements so that they all present a similar range of variation. In this case, a low-resolution ADC with a fixed measurement range can be used.

4.5.2 Implementation on the case study

Product programming

To emulate the operation of the circuit within an application environment, we use an evaluation board where the circuit is mounted. Alongside the evaluation board, a stand alone debug probe is used to download and debug the firmware through a JTAG interface. The connection to the IDE through a PC is established by using a USB cable which also powers the debug probe. All the needed tools and software are provided by NXP Semiconductors. Figure 4.18 illustrates the experimental setup that we used to conduct the full experiment. A code dedicated to the on-line performance monitoring process has been developed. The flowchart of this code is illustrated in Figure 4.19.

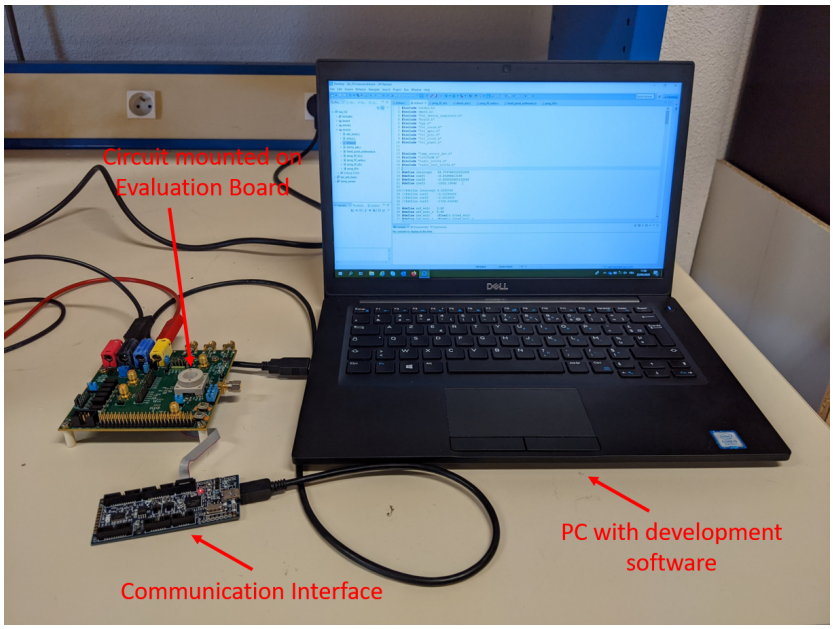


Figure 4.18: Illustration of the experimental setup

The code is mainly divided into three different stages, (*i*) the initialization stage, (*ii*) the measurement and storage stage and finally (*iii*) a computation stage. During the first stage we initialize the circuit and launch a test sequence that activates the transmitter. Once the circuit is ready, we loop through the selected indirect measurements to measure their raw value using the ADC and store them in the available SRAM memory. Then alongside the raw measurements of the different IMs, the model coefficients are fetched from the Flash memory and used to produce a power level prediction. Finally, we use the retained power level value under the nominal configuration, from the Flash memory, to calculate the power level variation. Therefore, if the variation in the power level is substantial, a flag is raised in order to indicate a performance deterioration.

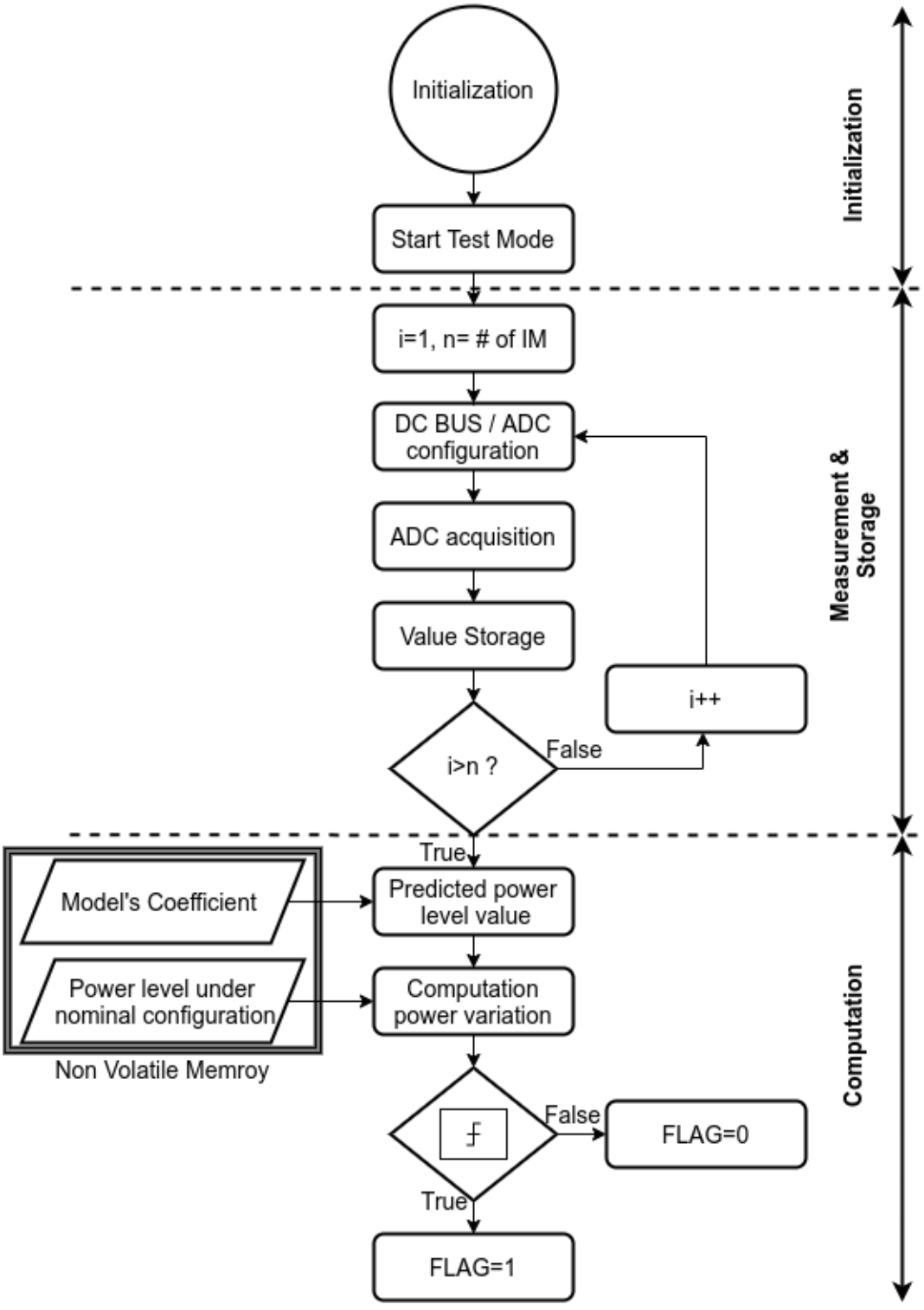


Figure 4.19: Flowchart of the on-line performance monitoring process

Note that during the execution of the program a number of outputs that correspond to different stages have been included. In particular the raw measurements of the selected indirect measurements, the power level variation based on the predicted power level and the status of the alert flag. These different outputs are transferred to the PC via the USB interface ad can be extracted with the help of the IDE. Note that in the final version of the code, the only output will be the alert flag. However, for validation purpose, it is interesting to have access to the raw measurements of the indirect measurements and the value of the predicted power level alongside the power level variation. Indeed, it permits to compare the result of the computation performed on an external PC with the result of the embedded computation within the circuit.

Initial results

The developed code has been launched for two configurations of the product, i.e. the nominal configuration (C172) and a degraded one (C241). For each configuration, 100 runs have been performed in order to analyze the repeatability of the prediction results. Figure 4.20 illustrates the predicted power level degradation observed in each configuration, for the 100 runs.

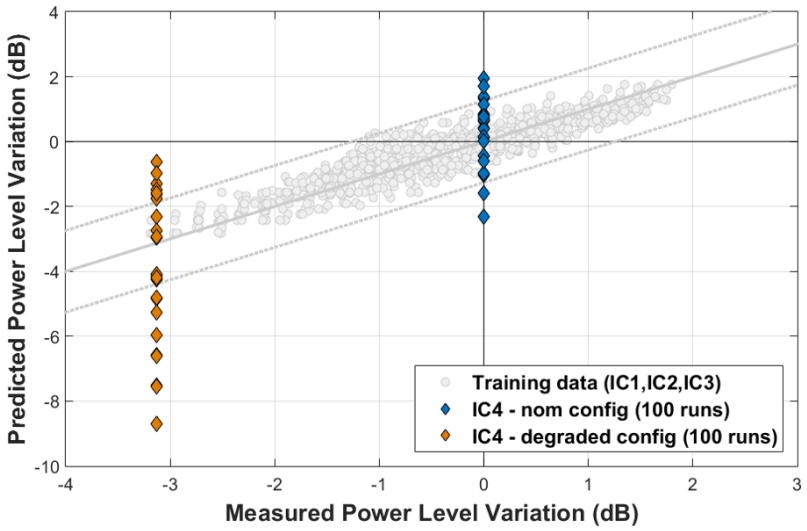


Figure 4.20: Embedded prediction of power level degradation on IC4

Results of Figure 4.20 reveal that there is a very large dispersion of the prediction results over the 100 runs, in both configurations, whereas the values of predicted power level degradation are expected to be contained in the range $[-1.26dB; +1.26dB]$ in the nominal configuration and $[-4.39dB; -1.87dB]$ on the degraded one. Obviously such a large dispersion does not permit to guarantee the certain detection of a power level degradation. We presume that this large dispersion comes from measurement repeata-

bility/accuracy problems.

Several elements support this assumption. First, DC35 and DC37, which are essential indirect measurements in the model, have a very low DC value in the range of few millivolts. Measurement of such small values is clearly sensitive to noise and therefore might be subject to a measurement repeatability issue. Moreover, the variation induced by the circuit configuration is also very small, i.e. a maximal deviation from the nominal value of only $0.34mV$. Correct evaluation of such a small deviation clearly necessitates high measurement precision. However, the ADC used for the embedded measurement can be programmed only for two different measurement ranges, i.e from 0 to $3.6V$ and from 0 to $0.9V$. This second measurement range is chosen for the measurement of DC35 and DC37. Taking into account the 12-bit resolution of the ADC, it means that one LSB correspond to $0.22mV$; a deviation of $0.34mV$ corresponds to a difference of 1.5 LSB, which means that only 2 bits out of the 12 bits of the ADC are actually exploited. Obviously, such a low equivalent resolution has a strong impact on the measurement accuracy and therefore on the prediction error, as discussed in the theoretical study of the ADC resolution presented in Section 4.5.1.

Use of averaging

A possible solution to cope with the problem of measurement repeatability and mitigate the impact of the low equivalent resolution of the ADC for the measurement of DC35 and DC37 is to implement averaging. This is a very common solution implemented in many instruments in order to improve the measurement accuracy.

A study has been conducted in order to determine which is the appropriate number of averaging that leads to a sufficient measurement accuracy. Practically, the initial measurement set that contains 100 values for the four indirect measurements involved in the regression model has been randomly re-sampled, varying the re-sample size. The re-sampling process has been iterated 1,000 times for each re-sample size. For each re-sampled set, the average of the DC measurements has been computed. Prediction is then performed using these averaged values as inputs of the regression model.

Results are summarized in Figure 4.21 that shows the boxplots of the predicted values over the 1,000 re-sampling iterations for different re-sample sizes (i.e. number of averaging), in the nominal and degraded configurations. By examining the results, it appears that the dispersion of the predicted values reduces when the re-sample size increases. Moreover, certain predicted values, especially for re-sample sizes of one and five, are situated far outside the expected range of each configuration, thus creating a crossover between the two regions. As previously stated, the aim is to avoid any crossover between the two regions, which is respected when at least ten samples are used to produce the predicted value. However, when using ten samples, the dispersion of the predicted value is adjacent to the boundaries of the expected range in the degraded

configuration. Thus, it is preferable to choose 20 as the number of applied averaging to predict the power level of the device.

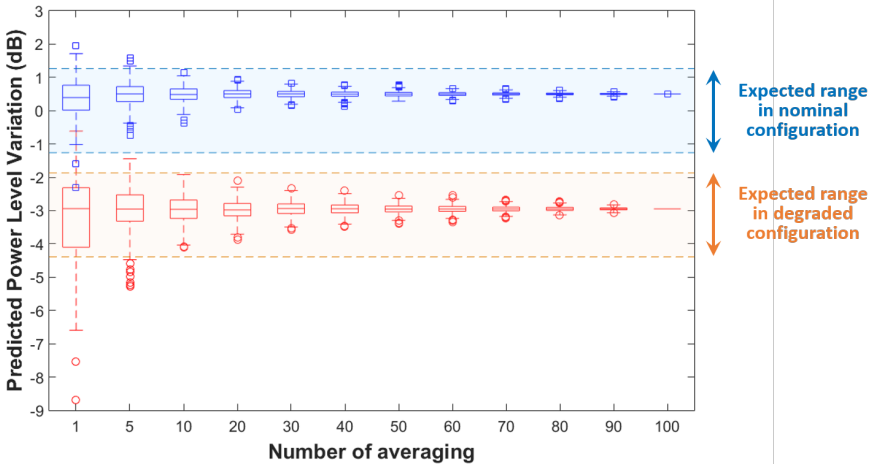


Figure 4.21: Boxplot of predicted values for power level degradation over the 1,000 re-sampling operation vs. the re-sample size, in the nominal and degraded configurations

Results

The initial code dedicated to the on-line performance monitoring process has been updated to include averaging on 20 measurements. The updated code has then been launched for the two configurations of the product, i.e. the nominal configuration (C172) and a degraded one (C241). In each configuration, the value of the predicted power level degradation has been transferred to the PC, as well as all the indirect measurement values collected over the 20 measurements. Using these indirect measurement values, we can therefore also perform the computation of the predicted power level degradation on the PC.

Results are illustrated in Figure 4.22, which shows the power level variation values predicted using either the original indirect measurements performed on the ATE or the embedded ones achieved within the circuit, in both configurations. Comparison between external computation performed on a PC and embedded computation achieved within the circuit is also provided. Many observations can be drawn out from the results in Figure 4.22. First of all, the difference in the predicted value between embedded and external computation is negligible. Indeed, the two points are superposed in both configurations and most importantly within the established boundaries ($\pm 3\sigma$ of prediction error). Besides, the original indirect measurements performed on the ATE produces a different prediction than the measurements achieved within the circuit, probably due to the difference in measurement accuracy.

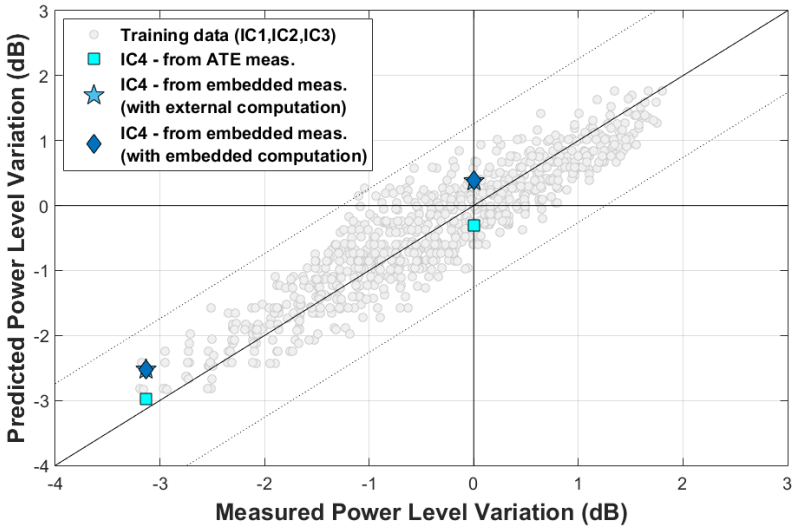


Figure 4.22: Comparison of power level degradation prediction on IC4 using either ATE or embedded measurements

Results presented in Table 4.4 show the prediction error across both the nominal and degraded configurations using either the original indirect measurements performed on the ATE or the embedded ones achieved within the circuit. Certainly there exists a prediction error in both methods due to the intrinsic imperfections of the established regression model. However, as previously stated, the difference in the measurement accuracy leads to a disparity in the prediction error. The prediction error observed when using the embedded indirect measurements is substantially higher than the one obtained with ATE measurements in the case of the degraded configuration, whereas it is roughly the same in terms of absolute value in the case of the nominal configuration. Nonetheless, the prediction error is acceptable enough for the proof-of-concept of an on-line performance monitoring based on an indirect test strategy.

Table 4.4: Summary of prediction error on IC4

		Nominal Configuration	**Degraded Configuration**
Prediction error using ATE indirect measurements (*dB*)		-0.311	+0.1246
Prediction error using embedded indirect measurements with averaging (*dB*)	**Embedded Computation within circuit**	+0.385	+0.602
	External computation on PC	+0.376	+0.602

4.6 Conclusion

In this chapter, we have investigated the feasibility of performing an on-line performance monitoring based on an indirect test strategy. We have introduced the strategy and proposed the essential requirements needed to adapt an indirect test strategy in order to monitor a performance on-line. The strategy has been implemented on a wireless microcontroller with the aim to monitor its transmitted power level using the available indirect measurements within the device. We have collected the different required measurements from four circuits using an industrial ATE while changing the configuration of two internal registers in order to expand our dataset and build a regression model.

Within the proposed strategy, we have investigated on the use of a limited set of non-linear transformations of the available indirect measurements in order to improve the accuracy of linear prediction models. Results have been presented, showing an improvement and achieving a sufficient accuracy for a proof-of-concept. With the help of the transformed indirect measurements we have established a regression model that uses three of the four circuits and validated the model on the fourth circuit. Moreover, we have analyzed the model capability to detect a performance deterioration by computing the power level variation between the retained value of the nominal configuration and the value predicted during its operation.

The presented on-line monitoring process has been applied to our case study. We have introduced an exploratory study on the effect of the ADC resolution on the performance of the prediction model. Subsequently, the practical implementation and the flowchart of the process have been presented. In addition, we have studied the importance of using averaging in the context of embedded measurements in order to produce a more reliable power level prediction. Once the averaging applied, the results in terms of prediction error have been presented under different circumstances (embedded vs external prediction) for only two important configurations (nominal and degraded). Results have shown that it is possible to produce a sufficient and reliable prediction using the available resources within our case study in order to detect a power level deterioration. These results can be considered as a valid proof that it is possible to monitor a performance on-line based on an indirect test strategy.

Conclusion

The work presented in this PhD manuscript introduces a generic methodology for the efficient implementation of an indirect test strategy. Considering the first challenge of improving the quality of an indirect test strategy, we introduced the concept of ensemble learning in order to improve the robustness and the efficiency of the implemented prediction model. The idea behind this procedure is that with an appropriate combination of various individual models, it should be possible to exploit the strengths and overcome the weaknesses of the individual models and achieve a better overall predictive performance. A full comparative study was conducted using different training set sizes. Results have shown that the use of Ensemble methods would enhance the overall performance of the prediction model across the different evaluation metrics. Moreover, we have shown that the performance of the classical prediction models can be met even when using a reduced set of learning instances if Ensemble methods are implemented. Finally, overall Stacking methods outperform all the different Ensemble techniques. Indeed, across the different specifications and training set sizes, Stacking performed better than all the other presented techniques. Nonetheless, this comparative study highlights a meaningful question in the context of an indirect test strategy, which is the pertinence of the metrics that are usually considered to evaluate the quality of a model, and the level of confidence in the test that we can have through these metrics. 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. In this context, we introduced an additional metric, called the trusted misclassification rate, that permits to evaluate the ability of a prediction model to perform correct classification while taking into account the conventional RF measurement uncertainty.

Following this first study, we have focused more specifically on test efficiency and confidence improvements. We have proposed a novel two-tier adaptive test flow approach based on a tolerance zone around the test limits in order to establish the confidence in the decision given by the indirect test. Initially, we presented the results on the efficiency of classical indirect test implementation, i.e. the tolerance zone equals to zero. Next, an optional one-dimensional filter which excludes circuits with extreme characteristics can be applied to the learning set. The results showed that the use of the optional filter would result in a weaker and an over-confident model in validation. Thus, in order to evaluate the two-tier adaptive test flow, we have chosen to use only

the prediction models established on the original Learning set. The results showed that a very good test quality can be preserved while achieving a substantial test cost reduction, i.e. a low misclassification rate of a few tenths of percents with less than 25% of the devices that need to go through a standard specification test in the case of a MARS model (respectively less than 31% for an SVM model). Using this methodology, test engineers have several choices at their disposal to ensure an efficient implementation of indirect testing.

Throughout this work, the indirect test strategy has been discussed as an alternative to the classical specification based testing for analog/RF integrated circuits. We have proposed an adapted strategy for the indirect test allowing to perform an on-line monitoring of the device performance in the final application. In this study, we have used a wireless microcontroller that has the required hardware capabilities to implement an on-line monitoring of the transmitted power level, based on the indirect test strategy. We collected test data from four integrated circuits while varying the configuration of two internal registers in order to create a substantial variation in the power level that emulates a performance degradation. Alongside the power level, the test data includes eleven indirect measurements. During the test data analysis, we verified that the observed variation on the power level was due to the different register configurations and not to the circuit variability. On the other hand, contradictory impressions were observed on the eleven indirect measurements: firstly, the eleven indirect measurements are divided into four different voltage ranges and secondly, only two indirect measurements show a substantial variation across the different register configurations. The situation is far from ideal, but it can be considered as sufficient to complete a proof-of-concept study. In addition, during this study we have examined the use of non-linear transformations on the original indirect measurements to increase the accuracy of linear prediction models. Once we have established the prediction model with the use of the non-linear transformed indirect measurements, we examined the effect of using an ADC for the digitization of the indirect measurements on the accuracy of the regression model. Finally, we have implemented our strategy on the device to predict the power level variation between two register configurations, namely the nominal configuration and a degraded configuration, in order to detect any performance level deterioration. The results have shown promising signs when the device is used to predict the power level variation, and the error is considered acceptable for a proof-of-concept case study.

This work opens interesting perspectives concerning indirect test strategies for Analog and RF integrated circuits. Further investigations may be conducted to improve the proposed flow by implementing others options in relation with feature selection, test metrics, the adaptive test flow and the embedded implementation of the on-line performance monitoring to further improve the quality of the model. Another direction is to study the impact of manufacturing process shift during the production test phase on the predictive models.

Bibliography

[1] H. S. Bennett, J. J. Pekarik, and M. Huang, "Radio frequency and analog/mixed-signal technologies for wireless communications," tech. rep., 2011.

[2] M. Dresler, "Technique to detect rf interface and contact issues during production testing," in *2006 IEEE International Test Conference*, pp. 1–6, IEEE, 2006.

[3] E. S. Erdogan and S. Ozev, "A multi-site test solution for quadrature modulation rf transceivers," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 9, pp. 1421–1425, 2011.

[4] P. N. Variyam and A. Chatterjee, "Enhancing test effectiveness for analog circuits using synthesized measurements," in *Proceedings. 16th IEEE VLSI Test Symposium (Cat. No. 98TB100231)*, pp. 132–137, IEEE, 1998.

[5] H.-G. Stratigopoulos and Y. Makris, "Nonlinear decision boundaries for testing analog circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 11, pp. 1760–1773, 2005.

[6] H.-G. Stratigopoulos, P. Drineas, M. Slamani, and Y. Makris, "Non-rf to rf test correlation using learning machines: A case study," in *25th IEEE VLSI Test Symposium (VTS'07)*, pp. 9–14, IEEE, 2007.

[7] H.-G. Stratigopoulos and Y. Makris, "Error moderation in low-cost machine-learning-based analog/rf testing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 2, pp. 339–351, 2008.

[8] P. N. Variyam, S. Cherubal, and A. Chatterjee, "Prediction of analog performance parameters using fast transient testing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, no. 3, pp. 349–361, 2002.

[9] M. J. Barragan, R. Fiorelli, G. Leger, A. Rueda, and J. L. Huertas, "Improving the accuracy of rf alternate test using multi-vdd conditions: Application to envelope-based test of lnas," in *2011 Asian Test Symposium*, pp. 359–364, IEEE, 2011.

[10] M. J. Barragan, G. Leger, and J. L. Huertas, "Multi-condition alternate test of analog, mixed-signal, and rf systems," in *2012 13th Latin American Test Workshop (LATW)*, pp. 1–6, IEEE, 2012.

[11] H. El Badawi, M. Comte, F. Azaïs, V. Kerzérho, S. Bernard, and F. Lefevre, "Which metrics to use for rf indirect test strategy?," in *2019 16th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, pp. 73–76, IEEE, 2019.

[12] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

[13] J. H. Friedman, "Multivariate adaptive regression splines," *The annals of statistics*, pp. 1–67, 1991.

[14] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[15] H.-G. Stratigopoulos, P. Drineas, M. Slamani, and Y. Makris, "Rf specification test compaction using learning machines," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 6, pp. 998–1002, 2009.

[16] S. Ellouz, P. Gamand, C. Kelma, B. Vandewiele, and B. Allard, "Combining internal probing with artificial neural networks for optimal rfic testing," in *2006 IEEE International Test Conference*, pp. 1–9, IEEE, 2006.

[17] S. S. Akbay and A. Chatterjee, "Built-in test of rf components using mapped feature extraction sensors," in *23rd IEEE VLSI Test Symposium (VTS'05)*, pp. 243–248, IEEE, 2005.

[18] L. Abdallah, H.-G. Stratigopoulos, C. Kelma, and S. Mir, "Sensors for built-in alternate rf test," in *2010 15th IEEE European Test Symposium*, pp. 49–54, IEEE, 2010.

[19] H.-G. Stratigopoulos, "Machine learning applications in ic testing," in *2018 IEEE 23rd European Test Symposium (ETS)*, pp. 1–10, IEEE, 2018.

[20] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[21] G. J. Székely and M. L. Rizzo, "Brownian distance covariance," *The annals of applied statistics*, pp. 1236–1265, 2009.

[22] M. J. Barragan and G. Leger, "Efficient selection of signatures for analog/rf alternate test," in *2013 18th IEEE European Test Symposium (ETS)*, pp. 1–6, IEEE, 2013.

[23] H. Ayari, F. Azais, S. Bernard, M. Comte, M. Renovell, V. Kerzerho, O. Potin, and C. Kelma, "Smart selection of indirect parameters for dc-based alternate rf ic testing," in *2012 IEEE 30th VLSI Test Symposium (VTS)*, pp. 19–24, IEEE, 2012.

[24] J. Liaperdos, A. Arapoyanni, and Y. Tsiatouhas, "Adjustable rf mixers' alternate test efficiency optimization by the reduction of test observables," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 9, pp. 1383–1394, 2013.

[25] A. Y. Ng, "Feature selection, l 1 vs. l 2 regularization, and rotational invariance," in *Proceedings of the twenty-first international conference on Machine learning*, p. 78, 2004.

[26] G. James, D. Witten, T. Hastie, and R. Tibshirani, "Linear model selection and regularization," in *An introduction to statistical learning*, pp. 203–264, Springer, 2013.

[27] G. Leger and M. J. Barragan, "Brownian distance correlation-directed search: A fast feature selection technique for alternate test," *Integration*, vol. 55, pp. 401–414, 2016.

[28] S. Larguech, F. Azaïs, S. Bernard, M. Comte, V. Kerzérho, and M. Renovell, "Efficiency evaluation of analog/rf alternate test: Comparative study of indirect measurement selection strategies," *Microelectronics Journal*, vol. 46, no. 11, pp. 1091–1102, 2015.

[29] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*. CRC press, 2012.

[30] M. J. Barragán, R. Fiorelli, G. Leger, A. Rueda, and J. L. Huertas, "Alternate test of lnas through ensemble learning of on-chip digital envelope signatures," *Journal of Electronic Testing*, vol. 27, no. 3, pp. 277–288, 2011.

[31] J. Wichard, M. Ogorzalek, and C. Merkwirth, "Entool-a toolbox for ensemble modelling," in *Europhysics conference abstracts ECA*, vol. 27, pp. A105–A105, EUROPEAN PHYSICAL SOCIETY, 2003.

[32] H. Ayari, F. Azais, S. Bernard, M. Comte, V. Kerzerho, O. Potin, and M. Renovell, "On the use of redundancy to reduce prediction error in alternate analog/rf test," in *2012 IEEE 18th International Mixed-Signal, Sensors, and Systems Test Workshop*, pp. 34–39, IEEE, 2012.

[33] S. Larguech, F. Azaïs, S. Bernard, M. Comte, V. Kerzérho, and M. Renovell, "A framework for efficient implementation of analog/rf alternate test with model redundancy," in *2015 IEEE Computer Society Annual Symposium on VLSI*, pp. 621–626, IEEE, 2015.

[34] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[35] B. Minasny and A. B. McBratney, "A conditioned latin hypercube method for sampling in the presence of ancillary information," *Computers & geosciences*, vol. 32, no. 9, pp. 1378–1388, 2006.

[36] P. Maxwell, "Adaptive testing: Dealing with process variability," *IEEE Design & Test of Computers*, vol. 28, no. 6, pp. 41–49, 2011.

[37] A. D. Singh and C. M. Krishna, "On optimizing vlsi testing for product quality using die-yield prediction," *IEEE transactions on computer-aided design of integrated circuits and systems*, vol. 12, no. 5, pp. 695–709, 1993.

[38] T. J. Powell, J. Pair, M. S. John, and D. Counce, "Delta iddq for testing reliability," in *Proceedings 18th IEEE VLSI Test Symposium*, pp. 439–443, IEEE, 2000.

[39] E. Yilmaz, S. Ozev, and K. M. Butler, "Adaptive test flow for mixed-signal/rf circuits using learned information from device under test," in *2010 IEEE International Test Conference*, pp. 1–10, IEEE, 2010.

[40] N. Kupp, P. Drineas, M. Slamani, and Y. Makris, "Confidence estimation in non-rf to rf correlation-based specification test compaction," in *2008 13th European Test Symposium*, pp. 35–40, IEEE, 2008.

[41] H. Ayari, F. Azais, S. Bernard, M. Comte, V. Kerzerho, O. Potin, and M. Renovell, "Making predictive analog/rf alternate test strategy independent of training set size," in *2012 IEEE International Test Conference*, pp. 1–9, IEEE, 2012.

[42] H.-G. Stratigopoulos and S. Mir, "Adaptive alternate analog test," *IEEE Design & Test of Computers*, vol. 29, no. 4, pp. 71–79, 2012.

[43] H.-G. Stratigopoulos, S. Mir, E. Acar, and S. Ozev, "Defect filter for alternate rf test," in *2010 15th IEEE European Test Symposium*, pp. 265–270, IEEE, 2010.

[44] G. Leger, "Combining adaptive alternate test and multi-site," in *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 1389–1394, IEEE, 2015.

[45] E. Yilmaz, S. Ozev, and K. M. Butler, "Adaptive multidimensional outlier analysis for analog and mixed signal circuits," in *2011 IEEE International Test Conference*, pp. 1–8, IEEE, 2011.

[46] S. Larguech, F. Azaïs, S. Bernard, M. Comte, V. Kerzerho, and M. Renovell, "A generic methodology for building efficient prediction models in the context of alternate testing," in *2015 IEEE 20th International Mixed-Signals Testing Workshop (IMSTW)*, pp. 1–6, IEEE, 2015.

[47] H.-G. Stratigopoulos and Y. Makris, "An adaptive checker for the fully differential analog code," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 6, pp. 1421–1429, 2006.

[48] S. K. Devarakond, S. Sen, A. Banerjee, V. Natarajan, and A. Chatterjee, "Built-in performance monitoring of mixed-signal/rf front ends using real-time parameter estimation," in *2010 IEEE 16th International On-Line Testing Symposium*, pp. 77–82, IEEE, 2010.

[49] J. Altet, D. Mateo, and D. Gómez, "On line monitoring of rf power amplifiers with embedded temperature sensors," in *2012 IEEE 18th International On-Line Testing Symposium (IOLTS)*, pp. 109–113, IEEE, 2012.

[50] D. Chang, S. Ozev, B. Bakkaloglu, S. Kiaei, E. Afacan, and G. Dundar, "Reliability enhancement using in-field monitoring and recovery for rf circuits," in *2014 IEEE 32nd VLSI Test Symposium (VTS)*, pp. 1–6, IEEE, 2014.

[51] D. Maliuk, H.-G. Stratigopoulos, H. Huang, and Y. Makris, "Analog neural network design for rf built-in self-test," in *2010 IEEE International Test Conference*, pp. 1–10, IEEE, 2010.

[52] L. Abdallah, H.-G. Stratigopoulos, and S. Mir, "True non-intrusive sensors for rf built-in test," in *2013 IEEE International Test Conference (ITC)*, pp. 1–10, IEEE, 2013.

[53] D. Maliuk and Y. Makris, "An analog non-volatile neural network platform for prototyping rf bist solutions," in *2014 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 1–6, IEEE, 2014.

[54] A. Dimakos, H.-G. Stratigopoulos, A. Siligaris, S. Mir, and E. De Foucauld, "Built-in test of millimeter-wave circuits based on non-intrusive sensors," in *2016 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 505–510, IEEE, 2016.

[55] M. Andraud, H.-G. Stratigopoulos, and E. Simeu, "One-shot non-intrusive calibration against process variations for analog/rf circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 11, pp. 2022–2035, 2016.

[56] F. Cilici, G. Leger, M. J. Barragan, S. Mir, E. Lauga-Larroze, and S. Bourdcl, "Efficient generation of data sets for one-shot statistical calibration of rf/mm-wave circuits," in *2019 16th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, pp. 17–20, IEEE, 2019.

[57] B. Halak, *Ageing of Integrated Circuits: Causes, Effects and Mitigation Techniques*. Springer, 2019.

[58] E. Maricau and G. Gielen, *Analog IC reliability in nanometer CMOS*. Springer Science & Business Media, 2013.

[59] P.-I. Mak and R. P. Martins, "High-/mixed-voltage rf and analog cmos circuits come of age," *IEEE Circuits and Systems Magazine*, vol. 10, no. 4, pp. 27–39, 2010.

[60] F. Horn, R. Pack, and M. Rieger, "The autofeat python library for automated feature engineering and selection," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 111–120, Springer, 2019.

[61] J. Shao, "Linear model selection by cross-validation," *Journal of the American statistical Association*, vol. 88, no. 422, pp. 486–494, 1993.

Personal Publications

Peer-reviewed scientific journals

[i] H. El Badawi, F. Azais, S. Bernard, M. Comte, V. Kerzerho and F. Lefevre, « Investigations on the Use of Ensemble Methods for Specification-Oriented Indirect Test of RF Circuits », Journal of Electronic Testing, 189-203 (2020)

International conferences with proceedings and reading committee

[ii] H. El Badawi, F. Azais, S. Bernard, M. Comte, V. Kerzerho and F. Lefevre, « Use of ensemble methods for indirect test of RF circuits: can it bring benefits? », *IEEE Latin-American Test Symposium (LATS), Santiago, Chile, 2019*

[iii] H. El Badawi, F. Azais, S. Bernard, M. Comte, V. Kerzérho, F. Lefevre, « Which metrics to use for RF indirect test strategy? », *International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), Lausanne, Switzerland, 2019*

[iv] H. El Badawi, F. Azais, S. Bernard, M. Comte, V. Kerzerho, F. Lefevre and I.Gorenflot, « Implementing indirect test of RF circuits without comprising test quality: a practical case study », *IEEE Latin-American Test Symposium (LATS), Maceio, Brazil, 2020*

[v] F. Azais, S. Bernard, M. Comte, B. Deveautour, S. Dupuis, H. El Badawi, M.-L. Flottes, P. Girard, V. Kerzerho, L. Latorre, F. Lefèvre, B. Rouzeyre, E. Valea, T. Vayssade, A. Virazel. « Development and Application of Embedded Test Instruments to Digital, Analog/RFs and Secure ICs ». *Proc. IEEE International Symposium on On-Line Testing And Robust System Design (IOLTS)*, Special Session 4, 1-4, 2020.

National conferences with proceedings and reading committee

[vi] H. El Badawi, F. Azais, S. Bernard, M. Comte, V. Kerzérho and F. Lefevre, « The use of ensemble methods for indirect test of RF circuits », *Colloque GDR SoC-SoC19, Montpellier, France, 19-21 juin 2019*

International conferences with restricted proceedings

[vii] H. El Badawi, F. Azais, S. Bernard, M. Comte, V. Kerzerho and F. Lefevre, « The Use of Ensemble Learning in Indirect Testing of Analog and RF Integrated Circuits », *South European Test Seminar (SETS), Pitztal, Austria, 2019*

Abstract

Process variations and physical defects can degrade the performance of a circuit, or even drastically affect its operation. It is therefore essential to verify the performance of each circuit produced in order to ensure the quality of manufactured devices shipped to the customers. This is the role of the testing process. This process represents a significant part of the total cost of an Integrated Circuit (IC), especially for analog and Radio-Frequency (RF) circuits, whose performances must be measured with sophisticated and expensive test equipment, following time-consuming test procedures. In order to reduce testing costs, an attractive solution is to adopt an indirect test strategy, which consists in measuring parameters that require only low-cost test resources and correlating these measurements, called Indirect Measurements (IMs), with the device specifications. This correlation is generally established using machine-learning algorithms during an initial learning phase. Then, during the production testing phase, every new device is evaluated using only the low-cost indirect measurements. While the indirect test strategy seems attractive, its deployment in an industrial context is viable only if sufficient test quality can be achieved. In this thesis, we have developed a methodology that permits to assist and guide the test engineer in its practical choices for an efficient implementation. Different aspects have been explored, such as the use of different types of regression models, the definition of pertinent metrics to evaluate the test efficiency, or the proposition of an original adaptive test flow in order to make a trade-off between test quality and test cost. We have also proposed an adaptation of the indirect test strategy allowing to perform on-line monitoring of a RF device performance within its application. All the results presented in this thesis have been evaluated using industrial test data on various case studies, which fully supports the developed innovations.

Résumé

Les variations de processus de fabrication et les défauts physiques peuvent dégrader les performances d'un circuit, voire affecter considérablement son fonctionnement. Il est donc essentiel de vérifier les performances de chaque puce fabriquée afin de garantir la qualité des circuits envoyés aux clients. C'est le rôle du processus de test. Ce processus représente une part importante du coût total d'un circuit intégré, en particulier pour les circuits analogiques et Radio-Fréquences (RF), dont les performances doivent être mesurées à l'aide d'un équipement de test sophistiqué et coûteux tant à l'achat qu'en temps d'utilisation. Afin de réduire les coûts de test, une solution intéressante consiste à adopter une stratégie de test indirect, qui consiste à mesurer des paramètres ne nécessitant que des ressources de test peu coûteuses, et à corréler ces mesures indirectes avec les performances du circuit. Cette corrélation est généralement établie à l'aide d'algorithmes d'apprentissage, au cours d'une phase d'apprentissage initiale. Ensuite, pendant la phase de test de production, chaque nouveau circuit est évalué en utilisant uniquement les mesures indirectes peu coûteuses. Si cette stratégie de test indirect semble attrayante, son déploiement dans un contexte industriel n'est viable que si la qualité des tests est suffisante. Dans cette thèse, nous avons développé une méthodologie qui permet d'assister et de guider l'ingénieur de test dans ses choix pratiques pour une mise en œuvre efficace. Différents aspects ont été explorés, tels que l'utilisation de différents types de modèles de régression, la définition de métriques pertinentes pour évaluer l'efficacité des tests, ou la proposition d'un flot de test adaptatif original permettant de réaliser un compromis entre la qualité et le coût du test. Nous avons également proposé une adaptation de la stratégie de test indirect en vue d'un contrôle en ligne des performances d'un dispositif RF dans son application. Tous les résultats présentés dans cette thèse ont été évalués en utilisant des données de tests industrielles sur différents circuits RF, soutenant pleinement les innovations développées.