



Confident alternate test implementation

Syhem Larguech

► To cite this version:

Syhem Larguech. Confident alternate test implementation. Micro and nanotechnologies/Microelectronics. Université Montpellier, 2015. English. NNT : 2015MONTS185 . tel-02052844

HAL Id: tel-02052844

<https://theses.hal.science/tel-02052844>

Submitted on 28 Feb 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de
Docteur

Délivré par l'Université de Montpellier

Préparée au sein de l'école doctorale **I2S**
Et de l'unité de recherche **LIRMM/UMR 5506**

Spécialité: **Systèmes Automatiques et Microélectroniques (SYAM)**

Présentée par **Syhem LARGUECH**

**Test indirect des circuits analogiques et
RF: Implémentation sûre et efficace**

Confident alternate test implementation

Soutenue le 03-12-2015 devant le jury composé de

Dr. Serge BERNARD	CR	CNRS-LIRMM	Invité (Directeur de thèse)
Dr. Florence AZAIS	CR	CNRS-LIRMM	Invitée (Co-directrice de thèse)
Dr. Haralampos STRATIGOPOULOS	CR	CNRS-LIP6	Rapporteur
Dr. Gildas LÉGER	CR	CSIC-IMSE	Rapporteur
Dr. Michel RENOVELL	DR	CNRS-LIRMM	Examineur
Dr. Laurent LATORRE	Pr	Univ. Montpellier	Examineur
Dr. Emmanuel SIMEU	Mdc	Univ. Grenoble	Invité
Dr. Manuel BARRAGAN	CR	CNRS-TIMA	Invité
Dr. François LEFÈVRE	Ing	NXP Semiconductors	Invité
Dr. Mariane COMTE	Mdc	Univ. Montpellier	Invitée
Dr. Vincent KERZERHO	CR	CNRS-LIRMM	Invité

Acknowledgements

I would like to thank Dr. Haralampos Stragopoulos, Dr. Gildas Leger, Dr. Emmanuel Simeu, Dr. Manuel Barragan, Dr. Michel Renovell, Dr. Laurent Latorre and Mr. François lefèvre for their implication. It is a pleasure and honor that they accepted to be part of my dissertation committee.

Acknowledge-
 the best team ever ♡
 Kerzerho, M. Comte and M.
 thesis. I really enjoyed the work
 each one of them. I am lucky to have many advisors by my side providing me
 with an excellent guidance and atmosphere for the research. I m deeply grateful
 to ♡ Florence who has always been present and effectively supporting this thesis;
 her valuable suggestions and writing skills have made many publications possible.
 ♡ Serge has always been supporting and encouraging me. I also want to greet his
 sense of humanity that keeps me feeling that he is a friend of mine. ♡ Vincent has
 been actively participating in this work. I want to thank him a lot. ♡ Mariane
 is the first one to believe in me since my masters till my thesis project. I thank
 her for her support, kindness and attitude. ♡ Michel is always with a smiling
 face, giving me energy and happiness. I want to thank him for his support
 and care. Also, I would like to thank all the ♡ LIRMM ♡ community and
 above all Arnaud Virazel and Pascal Nouet for hosting and supporting
 me during these 4 years. I m thankful to ♡ my colleagues and ♡ dear
 freinds Anu, Patcharee, Mohamed, Aymen, Alejandro, Stephane
 ... for the love, care and happiness that they give me.
 Finally, I m thankful to ♡ my father, my two broth-
 ers, my aunt, my grand mothers and my uncles
 who are always supporting me wherever
 I am in need. This work is dedi-
 cated to the memory of my
 mother, ♡ and all
 of you!
 ♡

Abstract



BEING able to check whether an IC is fully functional or not after the manufacturing process, is very difficult. Particularly for analog and Radio Frequency (RF) circuits, test equipment and procedures required have a major impact on the circuits cost. An interesting approach to reduce the impact of the test cost is to measure parameters requiring low-cost test resources and correlate these measurements, called indirect measurements, with the targeted specifications. This is known as alternate or indirect test technique because there is no direct measurement for these specifications, which requires so expensive test equipment and an important testing time, but these specifications are estimated w.r.t "low-cost measurements". While this approach seems attractive, it is only viable if we are able to establish a sufficient accuracy for the performance estimation and if this estimation remains stable and independent from the circuits sets under test.

The main goal of this thesis is to implement a robust and effective indirect test strategy for a given application and to improve test decisions based on data analysis.


To be able to build this strategy, we have brought various contributions. Initially, we have defined new metric developed in this thesis to assess the reliability of the estimated performances. Secondly, we have analyzed and defined a strategy for the construction of an optimal model. This latter includes a data preprocessing followed by a comparative analysis of different methods of indirect measurement selection. Then, we have proposed a strategy for a confident exploration of the indirect measurement space in order to build several best models that can be used later to solve trust and optimization issues. Comparative studies were performed on two experimental data sets by using both of the conventional and the developed metrics to evaluate the robustness of each solution in an objective way.

Finally, we have developed a comprehensive strategy based on an efficient implementation of the redundancy techniques w.r.t to the build models. This strategy has greatly improved the robustness and the effectiveness of the decision plan based on the obtained measurements. This strategy is adaptable to any context in terms of compromise between the test cost, the confidence level and the expected precision.

More generally, this study constitutes an overview to guide the test engineer regarding practical aspects of alternate test implementation.

Key words: Alternate test, indirect test, analog/RF ICs, data mining, machine learning, statistical techniques, correlation and modeling.

Résumé

 TRE en mesure de vérifier si un circuit intégré est fonctionnel après fabrication peut s'avérer très difficile. Dans le cas des circuits analogiques et Radio Fréquence (RF), les procédures et les équipements de test nécessaires ont un impact majeur sur le prix de revient des circuits. Une approche intéressante pour réduire l'impact du coût du test consiste à mesurer des paramètres nécessitant des ressources de test faible coût et corrélérer ces mesures, dites mesures indirectes, avec les spécifications à tester. On parle alors de technique de test indirect (ou test alternatif) car il n'y a pas de mesure directe des spécifications, qui nécessiterait des équipements et du temps de test importants, mais ces spécifications sont estimées à partir des mesures "faibles coûts". Même si cette approche semble attractive elle n'est fiable que si nous sommes en mesure d'établir une précision suffisante de l'estimation des performances et que cette estimation reste stable et indépendante des lots de circuits à traiter.

L'objectif principal de cette thèse est de mettre en œuvre une stratégie générique permettant de proposer un flot de test indirect efficace et robuste. Pour être en mesure de construire cette stratégie nous avons apporté différentes contributions. Dans un premier temps, on a développé une nouvelle métrique pour évaluer la robustesse des paramètres estimés. Dans un deuxième temps, on a défini et analysé une stratégie pour la construction d'un modèle optimal. Cette dernière contribution englobe un prétraitement de données puis une analyse comparative entre différentes méthodes de sélections de mesures indirectes ainsi que l'étude d'autres paramètres tels que la taille des combinaisons de mesures indirectes ainsi que la taille du lot d'apprentissage. Nous avons également proposé une stratégie d'exploration de l'espace des mesures indirectes afin de construire plusieurs modèles précis nécessaires pour résoudre les problèmes de précision et de confiance dans les estimations. Les études comparatives réalisées ont été effectuées sur deux cas d'études expérimentaux, en utilisant des métriques classiques ainsi qu'une nouvelle métrique permettant d'évaluer objectivement la robustesse de chaque solution.

Enfin, nous avons développé une stratégie complète mettant en œuvre des techniques de redondance de modèles de corrélation qui permettent d'améliorer clairement la robustesse et l'efficacité de la prise de décision en fonction des mesures obtenues. Cette stratégie est adaptable à n'importe quel contexte en termes de compromis entre le coût du test et les niveaux de confiance et de précision attendus.

Mots clés: Test alternatif, test indirect, circuits analogiques et Radio Fréquence, traitement de données, algorithme d'apprentissage, analyse statistique, modélisation, corrélation.

Contents

Acknowledgements	i
Abstract	iii
Résumé	v
List of Figures	ix
List of Tables	xiii
Introduction	1
1 Alternate test and data mining	3
1.1 Some principles of data mining	4
1.1.1 KDD Process	5
1.1.2 Machine-learning Algorithms	6
1.2 Indirect test	9
1.2.1 Introduction	9
1.2.2 Classification-oriented indirect test	10
1.2.3 Prediction-oriented indirect test	10
1.3 Prediction-oriented indirect test strategy	11
1.3.1 Training phase	13
1.3.2 Validation phase	13
1.3.3 Production testing phase	13
1.3.4 Problematic	14
1.4 Metrics for test efficiency evaluation	14
1.4.1 Accuracy metrics	15
1.4.2 Prediction reliability: Failing Prediction Rate (FPR)	16
1.5 Choice of the regression algorithm	18
1.6 Summary	22
2 Single model approach: outlier filtering and selection of indirect measurements	23
2.1 Outlier filtering	25
2.1.1 Exploratory IC space analysis	26

2.1.2	Adaptive k-filter	26
2.2	Methods for indirect measurement selection	30
2.2.1	IM selection based on Pearson correlation	32
2.2.2	IM selection based on Brownian distance correlation	33
2.2.3	IM selection based on SFS algorithm	34
2.2.4	IM selection using MARS built-in selection feature	36
2.3	Experimental setup for the evaluation of IM selection methods	36
2.3.1	IMs selection	37
2.3.2	Indirect test efficiency evaluation	38
2.4	Results and discussion	39
2.4.1	IM selection	39
2.4.2	Test efficiency	43
2.5	Summary	51
3	Multi-model approach: Model generation	53
3.1	IM space reduction	54
3.1.1	PCA-based reduction	54
3.1.2	Pearson correlation-based reduction	57
3.1.3	Iterative MARS-based reduction	58
3.1.4	Preliminary evaluation of IM space reduction solutions	61
3.2	Multi-model generation	62
3.2.1	Extended SFS-Parental strategy	62
3.2.2	Extended SFS-Non Parental strategy	63
3.2.3	Computational effort	64
3.3	Evaluation	65
3.3.1	Model accuracy: evaluation on TS	65
3.3.2	Built models evaluation on VS	68
3.3.3	Further analysis and discussion	74
3.4	Summary	78
4	Multi-model approach: Models redundancy	79
4.1	Model redundancy principle	80
4.2	Generic framework	83
4.2.1	Overview	83
4.2.2	Selection and construction of redundant models	84
4.2.3	Tradeoff exploration: reliability vs. cost	86
4.3	Results	87
4.3.1	Selection and construction of redundant models	87
4.3.2	Tradeoff between test cost and test reliability	90
4.4	Summary	94
	Conclusion	95
	Related publications	a
	Bibliography	c

List of Figures

1.1	KDD	5
1.2	Machine-learning algorithms	7
1.3	Classification-oriented testing	10
1.4	Prediction-oriented testing	11
1.5	Prediction-oriented alternate testing synopsis	12
1.6	Example of estimated vs. actual RF performance on the TS	16
1.7	Example of estimated vs. actual RF performance on the VS	17
1.8	An example of Failing Prediction Rate (FPR) achieved with two different models	18
1.9	PA test vehicle	19
1.10	RF transceiver test vehicle	20
2.1	Database form representation as a table of individuals (IC_k), attributes (IM_i) and classes (P_j)	24
2.2	Histogram distribution for an indirect measurement over the IC population showing outlier circuits	25
2.3	Examples of IM distribution encountered in the database (transceiver test vehicle)	26
2.4	Adaptive k-filter	28
2.5	Examples of IM distribution before (a) and after (b) the filtering process (transceiver test vehicle)	29
2.6	Correlation coefficient and estimation errors before and after the filtering process (transceiver test vehicle)	30
2.7	Correlation graph for a transceiver performance performed on training set (TS)	31
2.8	Curse of dimensionality issue in the context of indirect testing	32
2.9	The two main categories of feature selection algorithms	32
2.10	SFS search of an IM subset for prediction of one specification P_j	35
2.11	Experimental setup for test efficiency evaluation	37
2.12	Illustration of chosen IM subsets over 100 runs (transceiver test vehicle)	38
2.13	Model accuracy for the PA test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})	44
2.14	Model accuracy for the transceiver test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})	45
2.15	Prediction accuracy for the PA test vehicle considering different IM selection strategies and different sizes of training set	46

2.16	Prediction accuracy for the transceiver test vehicle considering different IM selection strategies and different sizes of training set	47
2.17	Prediction reliability for the PA test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})	48
2.18	Prediction reliability for the transceiver test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})	49
2.19	Comparative analysis of the different IM selection strategies for the two test vehicles	50
3.1	Pareto scaling graph for the first ten principal components	55
3.2	Correlation coefficient for MARS models built using an increasing number of IMs selected in the first PC	56
3.3	Scatter graph for model built using the 30 highest-ranked IMs in the first PC	56
3.4	Correlation coefficient computed on training set for models built on reduced IM space using PCA-based selection	57
3.5	IM-space reduction based on MARS built-in selection feature	58
3.6	Correlation coefficient for models built during iterative MARS-based selection	60
3.7	Correlation coefficient computed on training set for models built on reduced IM space using iterative MARS-based selection	61
3.8	Extended SFS-Parental strategy	62
3.9	Extended SFS-Parental strategy: IM combination lists	63
3.10	Extended SFS-non Parental strategy	64
3.11	Extended SFS-Non Parental strategy: IM combination lists	64
3.12	Model accuracy for the transceiver test vehicle	67
3.13	Model accuracy for the PA test vehicle	67
3.14	Prediction accuracy for the transceiver test vehicle	69
3.15	Prediction accuracy for the PA test vehicle	70
3.16	Prediction reliability for the transceiver test vehicle	73
3.17	Prediction reliability for the PA test vehicle	73
3.18	Influence of IM space reduction options on prediction reliability for the transceiver test vehicle	75
3.19	Influence of extended-SFS options on prediction reliability for the transceiver test vehicle	77
3.20	Influence of extended-SFS options on prediction reliability for the PA test vehicle	77
4.1	Comparison between prediction reliability results evaluated on training and validation sets	80
4.2	Two-tier alternate test synopsis with guard-band allocation	81
4.3	Two-tier alternate test synopsis with model redundancy	82
4.4	Procedure for confidence estimation based on model redundancy	83
4.5	Overview of the proposed generic framework	84
4.6	Meta-model construction with cross-validated committees (k=10)	85
4.7	Exploration of cost-reliability tradeoff, for a given IM subset size	86
4.8	Prediction reliability for models built with 3 IMs (models generated from extended-SFS)	88

4.9	Accuracy and reliability metrics for redundant models vs. size of selected IM subsets	89
4.10	Evaluation of different model redundancy implementations: (a) FPR vs. divergence threshold, (b) Retest vs. divergence threshold, (c) FPR vs. Retest . .	91
4.11	Prediction reliability achieved by different implementations of model redundancy, for 3 values of acceptable Restest level	92
4.12	Trade-off between test cost and test reliability for different sizes of selected IM subsets (minimum front of FPR obtained from the different scenarios of redundant model generation)	93

List of Tables

1.1	Test vehicle databases	15
2.1	Number of outliers eliminated from the PA database for different values of k .	27
2.2	Number of outliers eliminated from the transceiver database for different values of k	28
2.3	Test vehicle databases after outlier filtering	30
2.4	Selected IM subsets for the PA test vehicle according to different selection strategies, different training set sizes, and different values for the maximum number of IMs used to predict the performance	40
2.5	Selected IM subsets for the PA test vehicle according to different selection strategies, different training set sizes, and different values for the maximum number of IMs used to predict the performance	42
2.6	Computational time for the transceiver test vehicle	43
3.1	Evaluation of IM space reduction solutions	61
3.2	Rms training error of generated models for the transceiver test vehicle	66
3.3	Rms training error of generated models for the PA test vehicle	68
3.4	Rms prediction error of generated models for the transceiver test vehicle . . .	69
3.5	Maximal prediction error of generated models for the transceiver test vehicle .	70
3.6	Rms prediction error of generated models for the PA test vehicle	71
3.7	Maximal prediction error of generated models for the PA test vehicle	71
3.8	Failing prediction rate of generated models for the transceiver test vehicle . . .	72
3.9	Failing prediction rate of generated models for the PA test vehicle	72
4.1	Implemented scenarios for redundant model generation	85

Introduction

THE only viable solutions to test analog/RF integrated circuits are the specification-oriented ones. In other words, the test procedures have to estimate the device specifications for the pass/fail decision during the production phase. For RF and high performance analog devices, these procedures require expensive Automatic Test Equipment (ATE) with high-speed and high-precision analog/RF test resources. In addition to the direct ATE cost, the specific test environment increases the overall test cost as additional test facilities, test equipment maintenance and test development engineering [1].

Furthermore, because signal integrity is mandatory for RF measurement we have to consider RF probing issues, coaxial cable interfacing, matching functionality and board to device contact [2]. This context is particularly critical during the wafer test. In order to encounter these issues, manufacturers usually perform DC and low frequency measurements at the Wafer Test Level and focus on RF performances at the Package Test Level [1]. However, such kind of test increases also the test cost because the defective devices are identified in a backward level. In other words, the cost of packaging phase of the failed device is added to the overall test cost. Besides, the high cost of test is also due to the long time required to test analog/RF device. Indeed, the measurement of only one specification might be long and a large number of performances have to be measured. This is particularly true for some complex RF devices addressing RF multi-modes, which usually lead to an excessive test time. Moreover, as the analog/RF test resources are limited on RF testers, the multi-site test solutions are usually impossible for analog/RF devices [3].

Finally, because technologies scaling are following Moore's laws, the latest manufacturing technologies (i.e. System-On-a-Chip, System-In-Package, and Through Silicon Via 3D Integrated Circuits) offer very high density. In this context, it becomes impossible to access all the inner component's primary inputs and outputs in order to provide stimuli and monitor test responses.

Several cost-reduced RF IC testing strategies have been proposed in the literature to overcome the cost and inabilities of specification-based testing for analog and RF circuits. We can cite techniques based on analog Built-In-Self-Test (BIST) and Design for Testability (DFT) [4]. Other solutions rely on improving RF probing technologies and measurements accuracy [5]. In this context, the "alternate test" strategy (also called "indirect test") has appeared as a novel and promising strategy especially at Wafer Test Level [6] [7].

Alternate test offers several advantages compared to the conventional analog/RF test practices. The general goal of the alternate test strategy is to establish the correlation between

two data spaces: the "low-cost measurement space" (Indirect Measurement IM) and the "expensive specification space". Based on machine learning and data mining tools, indirect test principle assumes that we can use simple and low-cost analog/RF measurements to decide what is considered as pass/fail devices during the production phase. This approach has been applied to various types of analog and RF circuits.

The general purpose of this PhD report is to establish a framework for an efficient implementation of alternate test for analog/RF circuits. The target is not only the accuracy of performance estimation, but above all to 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 alternate test. After summarizing the main existing data-mining tools and approaches, we remind the steps to implement the alternate test approach referring to the data-mining process. Then, we present different metrics to be used for the evaluation of the alternate test efficiency. We also introduce a new metric called Failing Prediction Rate (FPR) which was developed to assess the built model reliabilities. Finally, we compare the performance of some learning algorithms on our datasets in order to choose the appropriate algorithm for our framework.

In the second chapter, we present a complete study in order to build a robust single model. We firstly introduce a basic filter to remove aberrant circuits. Then, we perform a comparative analysis of various IM selection strategies, which is an essential step for efficient implementation of this technique. The objective behind is to perform a robust strategy to build the best single correlation model.

In the third chapter, we develop two strategies for multi-model generation. The proposed strategies are based on a reduction phase of the explored IM-space. Comparative analysis on the proposed strategies and the techniques of the IM space reduction are then given.

In the fourth chapter, we present a generic framework for efficient implementation of alternate test. The propose implementation uses model redundancy. This involves an exploration of the tradeoff between cost and robustness. Furthermore in order to increase confidence, we have investigated an original option which consists in building the meta-models using ensemble learning.

Finally, the main contributions of this thesis are summarized in the conclusion and perspectives for future work are presented.

Alternate test and data mining

Contents

1.1 Some principles of data mining	4
1.1.1 KDD Process	5
1.1.2 Machine-learning Algorithms	6
1.2 Indirect test	9
1.2.1 Introduction	9
1.2.2 Classification-oriented indirect test	10
1.2.3 Prediction-oriented indirect test	10
1.3 Prediction-oriented indirect test strategy	11
1.3.1 Training phase	13
1.3.2 Validation phase	13
1.3.3 Production testing phase	13
1.3.4 Problematic	14
1.4 Metrics for test efficiency evaluation	14
1.4.1 Accuracy metrics	15
1.4.2 Prediction reliability: Failing Prediction Rate (FPR)	16
1.5 Choice of the regression algorithm	18
1.6 Summary	22

THE conventional practice for testing analog and RF circuits is specification-oriented, which relies on the comparison between the measured value and tolerance limits of the circuit performances. While this approach offers good test quality, it often involves extremely high testing costs. Indeed, the measurement of analog or RF performances requires dedicated test equipment which has to follow the continuous improvement in the performances of new ICs generation. It becomes difficult and very expensive to find the instruments to measure accurately the specifications. In addition to the high cost of Automated

Test Equipment (ATE), the nature of each individually measured performance may imply a repeated test setup which further increases conventional test time. Moreover, as design trends tend to integrate complex and heterogeneous systems in one package, new technical difficulties are added to the heavy test costs. For instance, it becomes impossible to access all the inner components primary inputs and outputs in order to provide stimuli and catch test responses. Finally, in the case of RF signals; a key challenge is to perform RF measurements at wafer level due to probing issues, and applying wafer-level specification-based testing at 100% is rather impossible. In this context, numerous research works can be found over the past twenty years on this topic. These usually try to overcome the cost and inabilities of specification-based testing for analog and RF circuits.

Towards RF test cost reduction, some research is designed to compact the number and types of specification tests that are operated within the production testing phase [8] [9]. Others have proposed a substitute solution namely "alternate test" (also called "indirect test") which has emerged as an attractive solution. The proposed solution relies on the power of machine learning and data mining tools to establish a simple and low-cost analog/RF specification test. The idea is to replace the conventional analog or RF performance measurements by some simple and low-cost measurements. The fundamental principle is actually based on exploiting the correlation between these two. This correlation is mapped through a nonlinear and complex function that can be determined by the mean of machine-learning algorithms. This correlation is then exploited during the mass production testing phase in order to deduce the circuit performances using only those low-cost indirect measurements. This approach has been applied to various types of analog circuits, including baseband analog [6] [7][10], RF [11][12][13][14], data converters [15][16], and PLLs [17].

This indirect test approach requires data mining tools and additional treatments integrated on a complete test board process in order to find the best ways for its implementation.

This chapter is organized as follows. Section 1.1 presents some principles of data mining. Section 1.2 presents the indirect test. Section 1.3 describes the prediction-oriented indirect test strategy. In section 1.4, we define the metrics that we will use with this work. Section 1.5 exposes a comparative study between 3 widely used regression algorithms in order to choose the appropriate machine-learning algorithm for our study. Finally, section 1.6 concludes the chapter.

1.1 Some principles of data mining

Data mining is the field that studies large data sets. The aim is to find models that can summarize big data in order to convert them later into information and then into knowledge [18]. More precisely, data mining algorithms aim to identify what is deemed knowledge according to the disposed features and try to extract the relevant patterns from data. Data mining is generally performed inside a multi-step process and it relies heavily on Machine Learning Algorithms (MLA).

1.1.1 KDD Process

The term Knowledge Discovery in Databases (KDD) refers to the board process of finding knowledge in data with the application of particular data mining methods. It is of interest to researchers in machine learning, pattern recognition, databases, statistics, artificial intelligence, knowledge acquisition for expert systems, and data visualization [19]. The unifying goal of the KDD process is to extract knowledge from data in the context of large databases. It involves the evaluation and possibly interpretation of the patterns to make the decision as to what qualifies as knowledge. It is a multi-step process that includes the choice of preprocessing, sampling, projections and data mining tools.

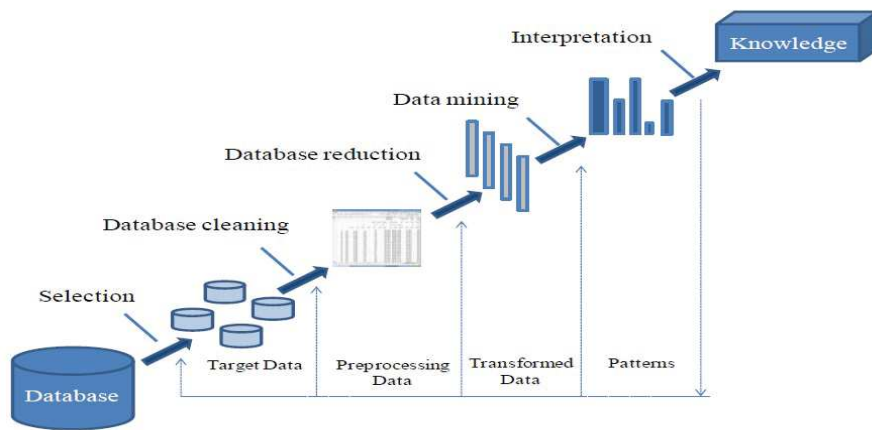


Figure 1.1: KDD

The KDD process involves the following steps:

- ◇ Selection: creating a target data set on which discovery will be performed after understanding of the application domain.
- ◇ Data cleaning and preprocessing: removal of noise or outliers, strategies for handling missing data fields, feature scaling...
- ◇ Data reduction and projection: using dimensionality reduction or transformation methods to reduce the dataset while keeping useful features to represent the data depending on the goal of the task.
- ◇ Data mining: deciding whether the goal of the KDD process is classification, regression, clustering, etc. Searching for the appropriate machine-learning algorithms and patterns of interest.
- ◇ Analyze discovered knowledge: interpreting mined patterns

The KDD process coheres with the alternate test context. It corresponds actually to the classical implementation of an indirect testing flow. In this manuscript, we will perform the

various described steps above with some adaptation to the alternate test context. Also, we will expose additional processing to achieve our objectives. More details on the indirect testing implementation will be found in the next sections.

1.1.2 Machine-learning Algorithms

Overview of Machine-Learning Algorithms

Machine-learning algorithms are one of the most exciting recent technologies. They are omnipresent in our daily life and we are using them unconsciously. Website engines like Google use MLA to search in internet. Facebook or Instagram applications use MLA to recognize our friend photos. Spam filters save us from thousands of spam emails using MLA. Machine learning then tries to mimic how human brain learns and investigated in different complex fields.

MLA was developed as a new capability for computers and today it touches many statements of industry and basic science [20][21]. Data mining is one of the fields based on MLA. For example, biologists are performing MLAs on data collected from genes and ADN sequences to understand the human geneses [22]. All fields of engineering as well are using MLAs, engineers have big datasets to understand using learning algorithms. Also, MLAs are used to perform applications that can't be programmed by hand as autonomies helicopter, where a computer learns by itself how to fly a helicopter based on MLAs [23].

Two kind of machine-learning algorithms exist: supervised and unsupervised machine-learning algorithms.

Figure 1.2 represents the two main families of MLA. For both graphs, X_1 and X_2 represent the features (also called attributes). We have considered in this example two features but obviously we can have more than 2. The two MLA families are:

- ◇ Supervised learning: is a learning problem where, for a given dataset, the features are labeled and the output space(class)is known. Two kinds of problems can be solved with supervised learning algorithms:

Regression problem: is one of the most popular statistical problems among the data mining community. Regression algorithms try to look for a law which connects something that is continuous in input with something that is continuous in output. As a regression problem example, housing price prediction where we want to predict the price of houses according to the house sizes.

Classification problem: consists in associating an individual X to a class C from K classes based on their types, properties and behaviors. The input space is divided in two or more classes. Example, classify patient as holder of a malignant tumor (class 1) or benign tumor (class 0) regarding the tumor size.

- ◇ Unsupervised learning: in this case, for a given data set, we don't know either data labels or to which group the data belongs and we want to find some structure in the data for clustering issues. This can make it hard to reveal the information contained in those data

sets. This problem is considered to be as an unsupervised problem and should be solved by an approach called Clustering [24]. As example, Google news use clustering algorithms every time we are searching for news on the Internet. It displays thousands of different pages related to the same news topic that we are searching for and where it clusters automatically the same topics.

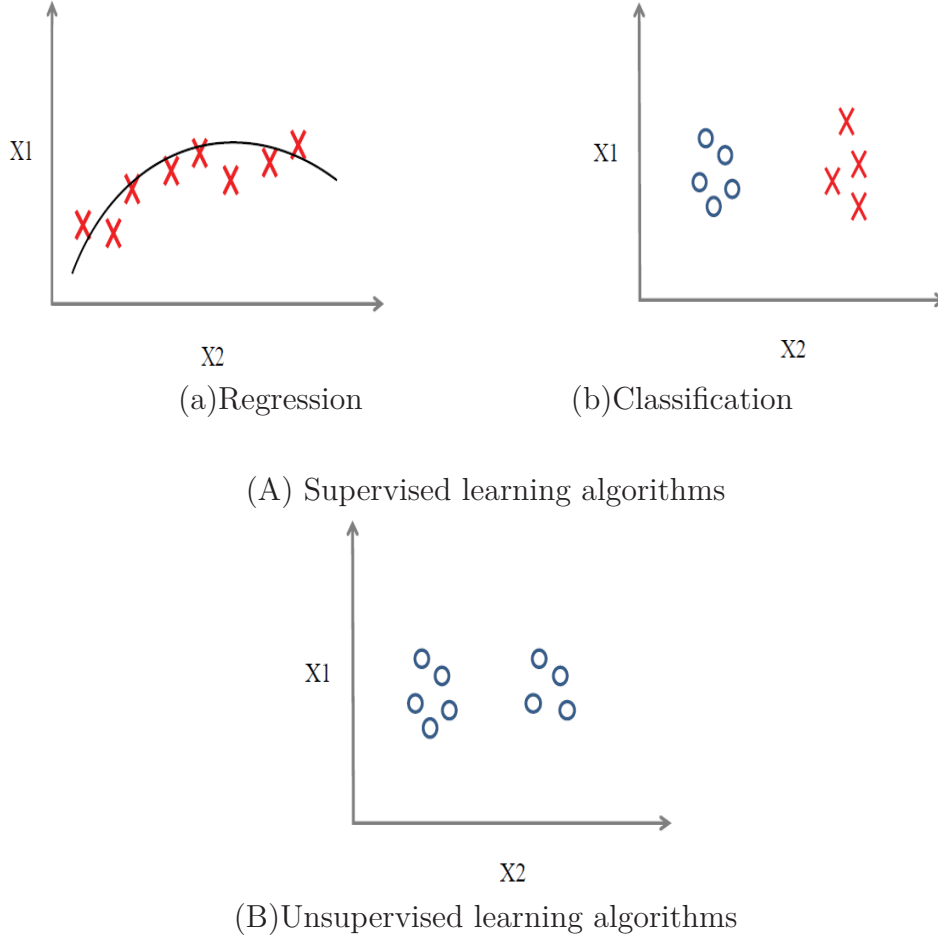


Figure 1.2: Machine-learning algorithms

In our study, we are in the supervised learning context as both input and output variables are defined. The next section focuses on two kinds of potential algorithms: regression and classification algorithms.

Regression Algorithms

There are several machine-learning algorithms used for regression mapping. In the same way, in the field of indirect testing, people used various algorithms [25]. We define below briefly three of the most commonly used learning algorithms:

- ◇ Artificial Neural Network (ANN) is a mathematical model inspired from biological neural networks, which involves a network of simple processing elements (neurons). ANN is a

multi-layered system composed of an input, hidden and output layers. Through the multiple layers of the network, a back propagation algorithm is used to adjust the parameters and threshold value of the network in order to minimize the error value for all inputs. Neural networks can be used for modeling complex relationships between inputs and outputs and they have been successfully implemented for prediction tasks related to statistical processes. Although ANN is very useful, it has some drawbacks. It is computationally expensive. Also, it is a black box learning approach: we cannot interpret relationships between inputs, layers and outputs. The difficulty in using ANN comes with the choice of the number of neurons. On the one hand, too few neurons lead to high training and generalization errors due to underfitting and high statistical bias. On the other hand, too many neurons lead to low training errors but high generalization error due to overfitting and high variance.

- ◇ Regression trees can be defined as a set of rules. It starts with all the data in one node and then splits it into two daughter nodes depending on the implemented rules. Each daughter node is then split again. This process is repeated on each derived daughter in a recursive manner. The recursion is completed when the subset at a node has all the same values of the target variable, or when splitting no longer adds value to the predictions [26]. One of the questions that arises in a regression trees algorithm is the optimal size of the final tree. A tree that is too large risks overfitting the training data and poorly generalizing to new samples. A small tree might not capture important structural information about the sample space.
- ◇ Multivariate Adaptive Regression Splines (MARS) is a series of local regressions stitched together to form a single function presented for the first time by J. Friedman in [27]. It can be considered as an adaptation of the regression tree. It is a multiply looped algorithm that has a spline function in the innermost loop.

The MARS model can be viewed in the following form:

$$\hat{f}(x) = c_0 + \sum_{k=1}^k c_i B_i(x) + \sum_{k=2}^k c_{i,j} B_{i,j}(x_i, x_j) + \sum_{k=3}^k c_{i,j,k} B_{i,j,k}(x_i, x_j, x_k) + \dots \quad (1.1)$$

where B corresponds to the basis function, c is the weighting coefficient, and c_0 is the constant intercept term. The model is a sum of a sum of all basis functions that involve one variable, two variables, three variables and so on. During the construction of the predictive model, there is a first forward phase in which a greedy algorithm is used to select basis functions, i.e. the algorithm iteratively adds reflected pairs of basis functions. For each pair of basis functions added, a model is built and its performance evaluated in terms of training error. At each iteration, the algorithm selects the pair of basis functions that gives the largest reduction of training error. There is then a backward phase in which the algorithm removes terms one by one, deleting the least effective term at each step (according to the GVC criterion), until a user-configurable limit of maximum allowed basis functions is reached (NBFmax). MARS models are most useful in high dimensional spaces where there is little substantive reason to assume linearity or a low level polynomial fit. They combine very flexible fitting of the relationship between independent and dependent variables with model selection methods that can sharply reduce the dimension of the model [28]. MARS is quite recommended for continuous data processing. We will base our study on this al-

gorithm. Reasons and detailed explanation will be found in the last section and the next chapter.

Classification Algorithms

Several classification algorithms exist to deal with classification problems. There are some algorithms that are assigned for classification problems as the K-NN, ZeroR, OneR...

Furthermore, other algorithms such as ANN and Support vector machine (SVM) can be used for classification issues.

- ◇ K-Nearest Neighbors (K-NN): the principle is to classify a new sample on its appropriate class. According to the chosen K value, the algorithm computes the distance between the K nearest neighbors (nearest individuals) and the new sample. The number of the nearest neighbors determines the class to where the new individual belongs. KNNs are fast and simple to implement. The big issue related to the use of K-NN is to find the optimal configuration (the appropriate number of classes).
- ◇ Support vector machine (SVM): is a supervised learning algorithm which can be used for classification or regression. SVM constructs separating hyperplane for the classes, and tries to find the hyperplane with the maximum margin between the classes. Samples on the margin are called the support vectors [29].

As some MLA matters with feature scaling (MLA that depends on distances and uses gradient descent)[30], generally data analysts perform during the data preprocessing step a data normalization. We have used the rescaling and standardization techniques for the "low-cost indirect measurements" and performances respectively. The corresponding formulas for the rescaling and standardization techniques are respectively:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1.2)$$

$$x' = \frac{x - \bar{x}}{\sigma(x)} \quad (1.3)$$

where x is the original feature vector, x' is the normalized value, \bar{x} is the mean of that feature vector, and σ is its standard deviation. Feature rescaling resizes the range of features to scale the range in $[0, 1]$ or $[-1, 1]$. Feature standardization makes the values of each feature have zero-mean (when subtracting the mean in the numerator) and unit-variance. Both methods are widely used for normalization in many machine learning algorithms (e.g., support vector machines, logistic regression, and neural networks).

1.2 Indirect test

1.2.1 Introduction

The underlying idea of alternate testing is that process variations that affect the conventional performance parameters of the device (individual) also affect non-conventional low-cost indi-

rect parameters. If the correlation between the indirect parameter space (attributes) and the performance parameter space (classes) can be established, then specifications may be verified using only the low-cost indirect signatures. However the relation between these two sets of parameters is complex and usually cannot be simply identified with an analytic function. Two main directions have been explored for the implementation of the indirect test; the classification and the prediction-oriented strategies.

Below, for the given 2D illustrations, we present the Indirect Measurement Space by $IM = [IM_1, IM_2, \dots, IM_m]$, the Circuit Performance Space by $P = [P_1, P_2, \dots, P_l]$ and the Specification Tolerance Limits by $Limits = [P_p, min, P_p, max]$.

1.2.2 Classification-oriented indirect test

The "classification-oriented strategy" was examined in many studies in the literature [9] [31] [32] [33] [34]. The principle is to classify devices as good or faulty (PASS/FAIL) without predicting its individual performance parameters. In such kind of study we evoke a classification problem where the specification tolerance limits are therefore part of the strategy. As figure 1.3 illustrates, the PASS/FAIL decision is fixed on the IM space. The specification tolerance limits are used only on the learning step in order to build the decision boundaries on the IM space. After that, the performance space is no more used and only the learned boundaries will serve to classify any new device into the class it belongs to.

On one hand, this strategy seems to be as fast as the PASS/FAIL decision is made on the IM space without turning back to the specification space and without verifying the RF performances of the new devices. On the other hand, this approach cannot offer diagnosis capability. The other drawback is the necessity to have the specification tolerance limits from an early phase (learning phase). Indeed, in mass production testing the specification limits may change.

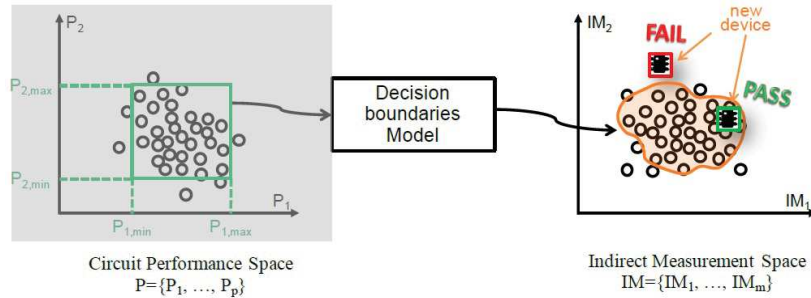


Figure 1.3: Classification-oriented testing

1.2.3 Prediction-oriented indirect test

The second strategy, the "prediction-oriented strategy", was adopted in many studies [7] [35] [36]. This strategy evokes a prediction problem where an estimation of the measured performances is provided by the end. Figure 1.4 illustrates the principle.

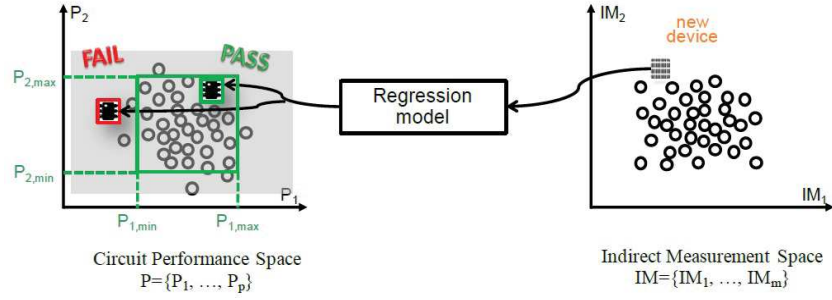


Figure 1.4: Prediction-oriented testing

The device performances are predicted instead of the decision boundaries by the regression model. On the prediction-oriented testing, the specification tolerance limits are known only on the production test phase unlike its counterpart "classification-oriented testing". The PASS/FAIL decision is made once the specification tolerance limits are provided.

This strategy has several advantages compared to the classification-oriented strategy. The main advantage is the potential use of the predicted specifications to adjust test production limits. Moreover, information about the predicted performances helps to diagnose, interpret and build confidence on the indirect test flow.

We have adopted the prediction-oriented strategy in our work due to these advantages. More details on the prediction-oriented testing will be found in the next section.

1.3 Prediction-oriented indirect test strategy

Figure 1.5 summarizes the prediction-oriented indirect testing synopsis, which involves three distinct phases: training, validation and mass production testing phases. Let $P = P_1, P_2, \dots, P_l$ denote the l performances of the Device Under Test (DUT) that need to be evaluated with the conventional specification prediction-oriented test approach, and $IM = [IM_1, IM_2, \dots, IM_m]$ a pattern of m low-cost indirect measurements. Note that the alternate test relies on the assumption that the DUT is affected by process variations but does not contain a hard defect. A defect filter such as the one proposed in [32] should therefore be included in the production testing phase in order to screen out circuits affected by hard defect before they are sent to the regression models.

One of the crucial issues associated with the indirect test approach consists in developing a model that satisfies the following challenges:

- ◇ High model accuracy: the regression model has to accurately represent the relationship between selected indirect parameters and specifications to be predicted.
- ◇ High prediction reliability: specifications have to be correctly predicted for all devices evaluated during the production test phase, although the model is built on a limited number of training instances.
- ◇ Low-cost model: the built model has to discount the test cost. The indirect test technique has to involve the minimal number of IMs as possible to ensure a low-cost test implementation.

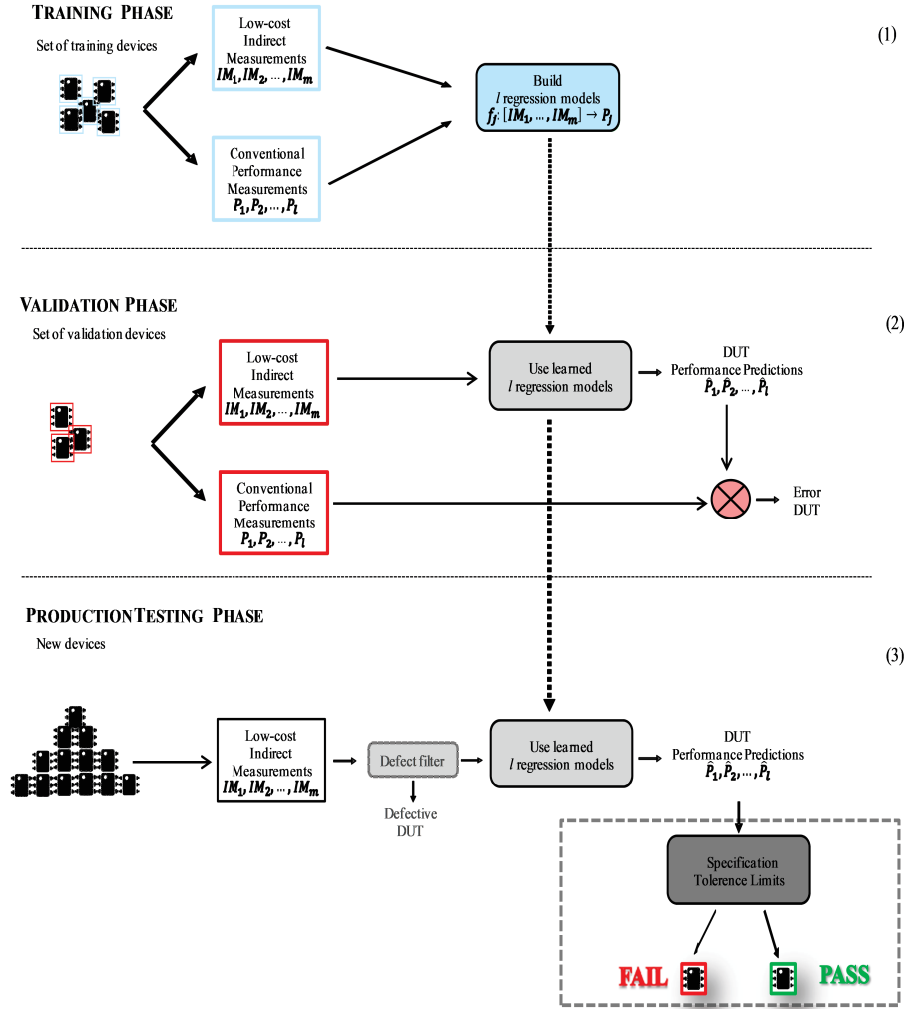


Figure 1.5: Prediction-oriented alternate testing synopsis

1.3.1 Training phase

During this phase, a set of devices so-called "Training Set" (TS) is assigned to feed a learning algorithm in order to build a regression model. The learning algorithm has two kinds of information as input: the low-cost indirect measurement dataset (IMs) and the conventional performance measurement (P) that were extracted from the TS. A regression model dedicated to the targeted performance is obtained as output. That means that for each performance parameter P_j , a machine-learning algorithm is trained over the two sets of measurements to build the regression model $f_j : [IM_1, \dots, IM_m] \rightarrow P_j$.

At this level, the model has to ensure a high accuracy; it has to accurately represent the relationship between selected indirect parameters and specifications to be predicted. The evaluation of the model accuracy is established by some specific metrics that we will introduce later on. Different regression tools can be employed, including polynomial regression, Multivariate Adaptive Regression Splines (MARS), Artificial Neural Networks (ANNs), support vector machines (SVMs), etc. Obviously, the quality of a regression model is strongly related to the choice of the input information (IMs) used to build the regression model. These indirect measurements have to be information-rich and correlate well with the device performances. Also, the appropriate choice of the learning algorithm to use will boost for sure the performance of the built model. Furthermore, the training devices and the training set size to consider theoretically have an impact on the built model. All these parameters have to be well fixed at this first phase of the indirect test synopsis. It is therefore also the objective of our work to analyze this aspect and more details will be given later on in the manuscript.

1.3.2 Validation phase

Following the training phase, a validation phase is executed. This phase is required to assess the built model efficiency. This evaluation is made on another set of devices so-called "Validation Set" (VS). The validation set disposes of the same kind of information as the training set but extracted from a different set of devices (VS).

Some classical metrics have been widely used in the literature to evaluate the prediction accuracy on VS. We will expose them and discuss their efficiency in the next paragraph. Moreover, we will introduce a new metric to assess the prediction reliability.

To achieve a viable efficiency assessment, the training and validation sets must have similar statistical properties. In this respect, we have used the LHCS "Latin Hyper-cube Sampling" process to divide the experimental dataset into two twin sets [37]. When the built regression model is evaluated in terms of prediction accuracy and reliability, we will be able to launch the production testing phase.

1.3.3 Production testing phase

In the production phase concerning a huge number of devices, we measure only one kind of information: the low-cost indirect measurements for each production device. The built regression model is then used to predict the corresponding RF performances based only on that information. After predicting the new device performances, we resort to the specification tolerance limits in order to classify the devices on PASS and FAIL classes.

1.3.4 Problematic

By adopting the prediction-oriented testing, crucial challenges have to be faced in order to establish a confident and efficient alternate test implementation. In fact, during the production testing phase, the high-priced performances are predicted by the mean of the "low-cost measurements" for a huge number of instances. Those predicted RF values suffer from a lack of confidence from industrials. The lack of confidence comes from different points. Dependency of prediction model with the training phase settings (the used IMs and TS) affects the model robustness and then the confidence in the strategy efficiency. Also, as the model is built on a limited number of training instances, an expression of a durable robustness is difficult face the set of production devices. Another point, many studies have shown that large prediction errors can be observed once implementing the prediction-oriented testing, which is not tolerated from industrials.

Those challenges can be exposed in the form of the following questions:

How to improve confidence in indirect test?

How to build an accurate and robust model?

Should we rely on a single model to predict all devices?

How to select pertinent IM subset(s)?

How to safely clean-up dataset(s)?

Which learning devices and which training set size should we consider?

How to make an efficient test and which test metrics to use?

How to manage the trade-offs between test cost and test reliability?

How to implement a generic and robust indirect test flow suitable for a given application?

Actually some of those problems have already been the focus of previous studies. We will try to answer all these questions in this manuscript along with proposing our contributions. Due to the limited time, other points were not included in the objectives of this work. They will be presented as further perspectives.

Two test vehicles fabricated by NXP Semiconductors will be used in this thesis to support the investigations and evaluate our propositions. The first one is a Power Amplifier (PA) for which we have production test data on a large set of devices, i.e. more than 10,000 devices. These data include 37 low-cost indirect measurements and one conventional RF performance measurement. The second test vehicle is a more complex device, i.e. an RF transceiver, for which we have experimental test data from a more limited set of devices, i.e. around 1,300 devices. These data include one conventional RF performance measurement and a large number of indirect measurements, i.e. 405 indirect measurements. Table 1.1 summarizes the composition of the databases provided by NXP, for the two test vehicles.

1.4 Metrics for test efficiency evaluation

A key issue for the deployment of the indirect test strategy is test efficiency evaluation. This latter is usually evaluated in terms of prediction accuracy and the most classical metric used in the literature is the average or rms error; maximal error is also sometimes reported. How-

Table 1.1: Test vehicle databases

Test vehicle	Number of circuits	Number of IMs	Number of measured Performances (Ps)
PA	11207	37	1
Transceiver	1299	405	1

ever these metrics do not give any information on prediction reliability, i.e. how many of the circuits are evaluated during the mass production test with a satisfying accuracy.

In this section, we first define the classical accuracy metrics and then, we introduce a new metric related to prediction reliability. These metrics will be used all along this manuscript for test efficiency evaluation.

1.4.1 Accuracy metrics

In this work, we will use as accuracy metrics the rms error (computed as the Normalized Root Mean Square Error *NRMSE*) and the maximal error, which are defined by:

$$\varepsilon_{rms} = \frac{1}{P_j | nom} \sqrt{\frac{1}{N_{dev}} \sum_{i=1}^{N_{dev}} (P(j, i) - \hat{P}(j, i))^2} \quad (1.4)$$

$$\varepsilon_{max} = \frac{\max(|P(j, i) - \hat{P}(j, i)|)}{P_j | nom} \quad (1.5)$$

where N_{dev} is the number of evaluated devices, $P(j, i)$ is the actual performance value of the i^{th} device, and $\hat{P}(j, i)$ is the predicted j^{th} performance value of the i^{th} device. These metrics are expressed in percentages through normalization with the j^{th} nominal performance value $P_j | nom$. The lower these metrics, the better the accuracy is.

Note that we can distinguish model accuracy and prediction accuracy depending on whether metrics are computed on the Training Set (TS) or on the Validation Set (VS)[38].

Model accuracy

Model accuracy is usually evaluated by computing ε_{rms} on training devices. In this case, the metric reveals the intrinsic model accuracy, i.e. the global ability of the model to accurately represent the correlation between the used indirect measurements and the device performance under consideration. Figure 1.6 shows a scatter plot example of estimated versus actual performance on the training set. This kind of graph gives a qualitative estimation of the model accuracy: a model is considered accurate when the learning devices (plotted with blue dots) follow the first bisector. The rms error gives a quantitative evaluation of how close the blue dots are to the first bisector.

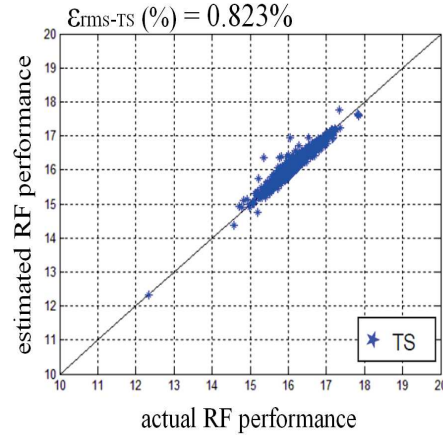


Figure 1.6: Example of estimated vs. actual RF performance on the TS

Prediction accuracy

Prediction accuracy is usually evaluated by computing ϵ_{rms} on devices of the validation set. In this case, the metric actually reveals the global ability of the model to accurately predict performance values for new devices. Figure 1.7 shows a scatter graph example of estimated versus actual performance on the training and validation sets (TS and VS). It can be observed that the large majority of validation devices (plotted with red dots) are accurately predicted, resulting in a low rms prediction error (in the same range as the rms error evaluated on TS). However it can also be observed that some devices exhibit a high prediction error. This is not expressed by the rms error since the number of these devices is extremely small compared to the number of devices correctly predicted; in contrast this is clearly expressed by the maximal prediction error. The maximal prediction error, which is not always reported in the literature, permits us to pinpoint this situation.

In our work, we evaluate both rms and maximal prediction errors.

1.4.2 Prediction reliability: Failing Prediction Rate (FPR)

The rms and maximal prediction errors allow us to quantify prediction accuracy but they are not sufficient to evaluate indirect test efficiency. Actually, many experiments reported in the literature on various devices demonstrate that very low average prediction errors can be achieved but this does not guarantee low maximal prediction error. This problem was highlighted in several studies [38] and [39]. Also, as evaluation is usually performed on a small set of validation devices (hundreds to some thousand instances) even if low maximal prediction error can be observed on a small validation set, there is no guarantee that the maximal prediction error will remain in the same order of magnitude when considering the large set of devices under test (several millions of devices). Finally it should be highlighted that the maximum prediction error is only an indicator of the worst case error, but gives no indication of the number of devices affected by a large prediction error.

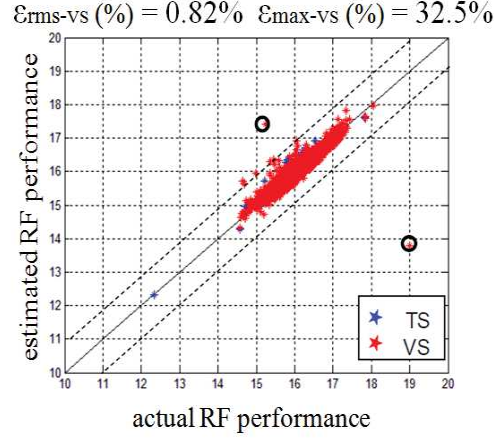


Figure 1.7: Example of estimated vs. actual RF performance on the VS

In this context we define a new metric dedicated to prediction reliability, which is related to the number of devices that are predicted with a satisfying accuracy for a given set of devices. More precisely this metric, called Failing Prediction Rate (FPR) [40], is defined as the ratio between the number of devices with a prediction error higher than a given ε_{meas} and the number of devices in the validation set, where ε_{meas} corresponds to the measurement repeatability error achieved when performing conventional measurement of the targeted circuit performance P_j .

The FPR, expressed in percentage, is given by:

$$FPR(\varepsilon_{meas}) = \frac{1}{N_{Vdev}} \sum_{i=1}^{N_{Vdev}} (|P(j, i) - \hat{P}(j, i)| > \varepsilon_{meas}) \quad (1.6)$$

where:

$$|P(j, i) - \hat{P}(j, i)| > \varepsilon_{meas} = \begin{cases} 1 & \text{if true,} \\ 0 & \text{otherwise.} \end{cases}$$

This metric permits us to quantify the indirect test efficiency with respect to the conventional test since it represents, for each circuit performance P_j to be evaluated, the percentage of circuits with a prediction error that exceeds the conventional measurement uncertainty. It is therefore a relevant metric to evaluate prediction reliability; it can also be used to compare the different Indirect Measurement (IM) selection strategies that we will introduce in the next chapter. As an illustration, figure 1.8 gives the FPR computed on the validation set for two models built with different combinations of indirect measurements. We can notice from this figure that the percentage of devices with a prediction error higher or equal to the conventional measurement uncertainty ε_{meas} resulting from Model 2 is less than the one given by Model 1; we can thus conclude that Model 2 offers better prediction reliability than Model 1.

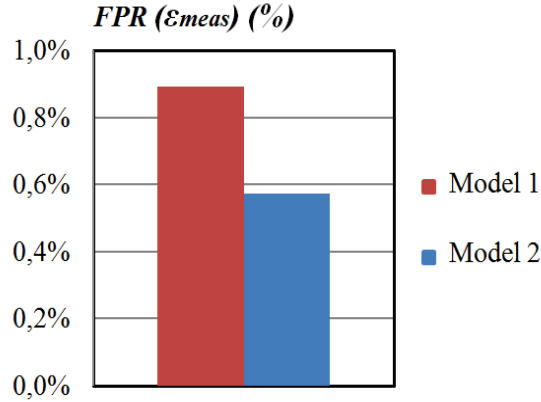
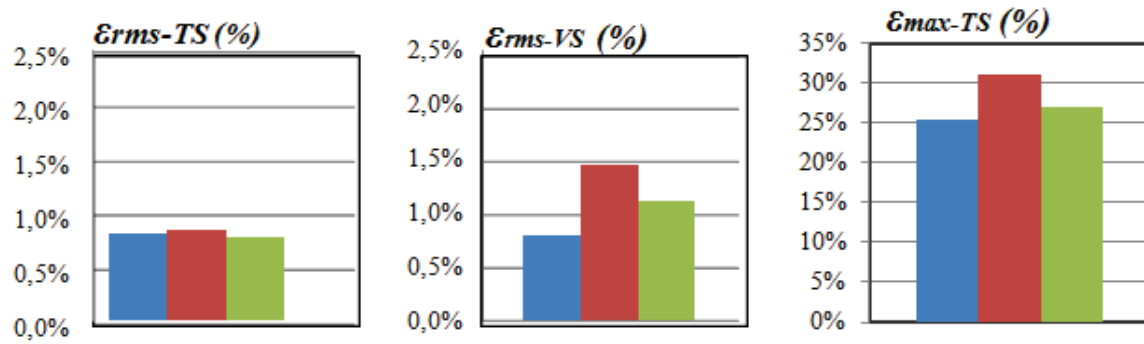


Figure 1.8: An example of Failing Prediction Rate (FPR) achieved with two different models

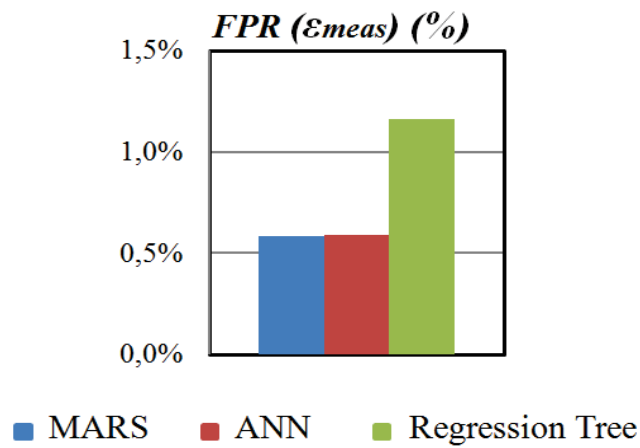
1.5 Choice of the regression algorithm

Various regression algorithms can be employed to build the regression model, including polynomial regression, Multivariate Adaptive Regression Splines (MARS), Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), etc. To motivate the choice of the appropriate machine-learning algorithm for our indirect testing flow, we have made a preliminary comparative study between 3 three widely used regression tools: the MARS, the ANN and the Regression Tree algorithms (implemented in ENTOOL Matlab-toolbox for regression modeling [41]). We can find brief first comparison studies on one of these algorithms efficiency in [42], but only in terms of accuracy (evaluated through Mean Square Error, MSE) on one case study. We extend here this study by comparing the performance achieved by the three algorithms not only in terms of accuracy but also in terms of reliability by exploiting the new FPR metric.

Practically for each test vehicle, we have selected a subset of 3 pertinent indirect measurements. These indirect measurements were provided to the different algorithms and a regression model was built for each MLA; Our analysis is based on two practical case studies: a Power Amplifier (PA) and an RF transceiver from NXP semiconductors. The assessment relies on the training and validation errors that were then computed on both test vehicles. Note that to ensure a meaningful comparison, we have used the same training and validation sets that were used for the 3 studied MLAs. Results are summarized in figures 1.9 and 1.10 for the PA and transceiver test vehicles respectively. Intrinsic Model model accuracy is evaluated by the rms error computed on the training set (ϵ_{rms-TS}), while prediction accuracy is evaluated by the rms and maximal prediction errors computed on the validation set (ϵ_{rms-VS} and ϵ_{max-VS}), and prediction reliability is assessed by the new failing prediction rate metric (FPR). Different remarks derive from those graphs.

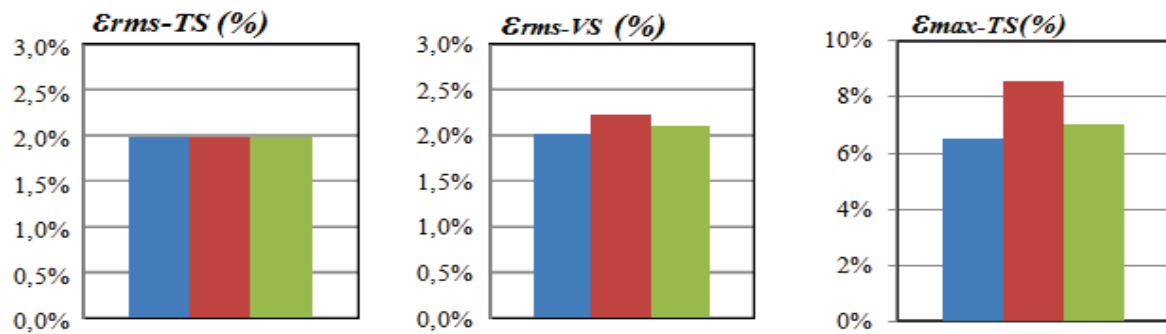


(a) Classical metrics

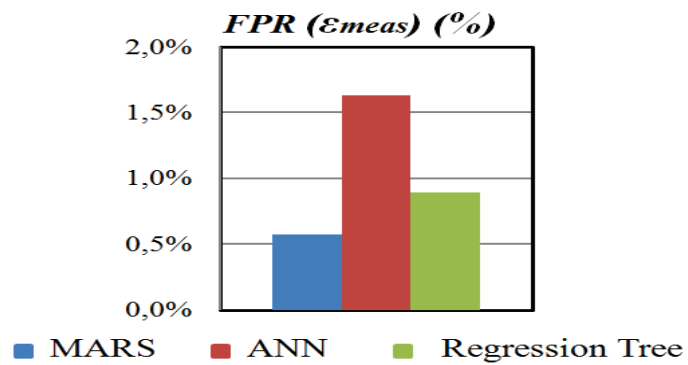


(b) FPR metric

Figure 1.9: PA test vehicle



(a) Classical metrics



(b) FPR metric

Figure 1.10: RF transceiver test vehicle

First regarding the classical metrics:

- the different MLAs lead to similar performance in terms of intrinsic model accuracy with equivalent rms errors on the TS;
- a modest advantage can be observed for the MARS algorithm in terms of prediction accuracy since it permits us to preserve the same rms error on the VS and exhibits the lowest maximal error. These results are in accordance with the previous study reported in [42].

Then regarding prediction reliability evaluated by the new metric, we can notice a clear advantage of the MARS algorithm over the two other algorithms, since it permits us to achieve the lowest FPR for both test vehicles. Based on these observations, we could conclude that the MARS algorithm leads to more reliable models than the ANN and the Regression Tree algorithms. From this study, we have decided to use the MARS algorithm to build regression models for the rest of our work.

1.6 Summary

Reducing the analog/RF IC test cost is a crucial issue in the semiconductors industry. There is a significant need to develop alternate techniques to the classical test approaches. Indirect test technique was proposed in order to lower the test cost. This approach is well known in the field of analog/RF IC testing but it suffers from a lack of confidence from industrials. Different steps to implement a robust indirect test flow are required. For this purpose, indirect testing as many research fields relies on the data mining tools and takes into account the KDD process in order to ensure the best exploration of the data space.

In the first part of this chapter, we have swiftly cited the different steps to implement the indirect test approach. We have mentioned some difficulties related to the application of the KDD process in our context. In fact, the effectiveness of a model is affected by several parameters. The case study and problem type are the key to identify the best data mining and MLA tools to use. Also, the data preprocessing step (as data cleaning and feature scaling) affect the prediction qualities. Another basic parameter is the right choice of training and validation sets. Those sets have to be split in a proper manner to equally present knowledge in data. Furthermore, other parameters can badly affect the model accuracy such as underfitting and overfitting problems. For this purpose data scientists refer to some estimation procedure (as cross-validation) in the case of very limited data sets or combine models to enhance the predictions qualities.

In the second part, we have announced our choice to work with the prediction-oriented indirect test strategy and we have exposed the different challenges that we have to deal with among this study. Also we have presented the different metrics used to evaluate the indirect test efficiency. Besides, we have introduced a new metric called FPR to assess the built model reliabilities. Finally, we have compared the performance of 3 commonly-used Machine-Learning Algorithms on our datasets and we have chosen to exploit the MARS algorithm for the implementation of the indirect test strategy.

In the next chapter we will come back to the different steps to build the prediction-oriented indirect test strategy. We will explain: the different steps, the different comparison analysis made and the additional processing to be integrated within the classical indirect test flow in order to develop a robust and generic indirect test implementation.

Single model approach: outlier filtering and selection of indirect measurements

Contents

2.1	Outlier filtering	25
2.1.1	Exploratory IC space analysis	26
2.1.2	Adaptive k-filter	26
2.2	Methods for indirect measurement selection	30
2.2.1	IM selection based on Pearson correlation	32
2.2.2	IM selection based on Brownian distance correlation	33
2.2.3	IM selection based on SFS algorithm	34
2.2.4	IM selection using MARS built-in selection feature	36
2.3	Experimental setup for the evaluation of IM selection methods	36
2.3.1	IMs selection	37
2.3.2	Indirect test efficiency evaluation	38
2.4	Results and discussion	39
2.4.1	IM selection	39
2.4.2	Test efficiency	43
2.5	Summary	51

BIG data is used to describe a massive volume of data. It is used in many fields: finance, marketing, biology, tourism... When dealing with such large amounts of data we face difficulties in being able to manipulate, and manage them. Computation time, storage difficulties volume, visualization and plotting big data represent serious issues for big data. In the context of indirect testing, the database contains a set of measurements, composed of m indirect measurements ($IM_i, i = 1, \dots, m$) and p performance measurements ($P_j, j = 1, \dots, p$), extracted from a population of n integrated circuits ($IC_k, k = 1, \dots, n$). The typical structure of the database is illustrated in figure 2.1: rows represent the individuals (integrated circuits,

IC_k) while the columns represent the attributes (indirect measurements, IM_i) and the classes (performance measurements, P_j) associated with the individuals. The objective is to explore the possible correlation between the attributes and the classes in order to establish a regression model for each class. These models will then be used to predict performance values of new individuals.

	IM1	IM2		...	IMj						P1	P2	...	
IC1														
IC2														
...														
ICi														

Figure 2.1: Database form representation as a table of individuals (IC_k), attributes (IM_i) and classes (P_j)

The regression model ensuring the correlation between the attributes and a given class is then the cornerstone for accurate prediction. Two main issues have to be faced for the construction of efficient regression models:

- ◇ The presence of outliers in the database: An outlier can be defined as "an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism" [43]. In the context of regression analysis, the problem with the presence of outliers in the database is that, since they are not consistent with the statistical nature of the remainder of data, they can affect the quality of the regression model. Therefore, they should be excluded from the database prior to the construction of the regression models.
- ◇ The curse of dimensionality: The curse of dimensionality refers to various phenomena that arise when analyzing and organizing data in high-dimensional spaces. In the context of regression analysis, it may seem interesting to have as many attributes as possible to maximize the opportunity to build a satisfactory model. However taking into account more attributes does not necessarily lead to a better model. In fact with a fixed number of training samples, the predictive power of a regression model reduces as the dimensionality increases, and this is known as the Hughes effect or Hughes phenomenon [44]. To avoid such an issue, regression models should involve an only limited number of attributes; selection methods are therefore required to identify the most pertinent attributes.

In this chapter, we address these two aspects. First we present an adaptive k-filter that permits us to remove outliers from the database. Then we investigate feature selection algorithms and more precisely we explore four different strategies for the selection of indirect measurements in order to avoid the curse of dimensionality issue. An experimental setup is developed to perform a comparative analysis of these strategies and results are evaluated on the two test vehicles for which we have production test data.

2.1 Outlier filtering

Alternate testing relies on the construction of a regression model that maps one device performance to some indirect measurements. It is well-established that such a regression function can be fit for data that are described by a fixed probability density function [45]. Outliers should therefore be excluded from the training phase since they are not consistent with the statistical characteristics of the remainder of the data.

In this work, we admit that outlier circuits are those whose measurement appears to deviate markedly from other circuits of the database. Generally outlier circuits correspond to circuits affected by random manufacturing defects, leading their measurement value to lie outside of the measurement statistical distribution. As an illustration, figure 2.2 shows the histogram distribution of an indirect measurement for the transceiver test vehicle. The indirect measurement of some circuits is so far from the mean of the IM distribution that they can be considered as outliers; those circuits should be identified and eliminated from the training dataset since they may adversely affect the quality of the learned regression model.

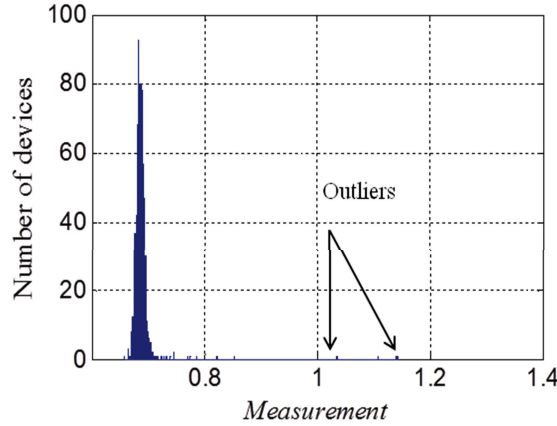


Figure 2.2: Histogram distribution for an indirect measurement over the IC population showing outlier circuits

Outlier detection has been studied for decades in a number of application domains such as fraud detection, activity monitoring, satellite image analysis or pharmaceutical research, but there is no single universally applicable technique [46]. In the context of alternate testing, a defect filter based on joint probability density estimation has been developed [32] to identify outliers that do not fit the expected multi-dimensional distribution of the indirect measurements. However this defect filter suffers from computational issue in the case of indirect measurement space with high-dimensionality. Another method consists in allocating non-linear guard-bands in the indirect measurement space [47]; however correct positioning of the guard-bands requires information on defective devices which is not always available. In this section, we develop a simple filter, so-called "adaptive k-filter" that can handle IM space of high-dimensionality and does not require information on defective devices. An exploratory analysis of the IC space is first presented in sub-section 2.1.1 for a better understanding of our database elements and a better setting of the filter requirements. The filter proposed for

screening out outlier circuits from the IC space is then exposed and tested in sub-section 2.1.2

2.1.1 Exploratory IC space analysis

First, we have performed an exploratory analysis of the IC space with the aim at getting familiar with the type of data and studying their characteristics. Indeed we have no information about the way that NXP has collected these data: what kind of indirect measurements, on which equipment and with which uncertainty, etc.

This analysis has revealed the diversity of data. For illustration, figure 2.3 shows some examples of IM distribution encountered in the database. For some IMs, the distribution is contained in a very narrow range while for others the distribution covers a wider range. The distribution can be either continuous or discrete, that latter probably related to the finite resolution of the test equipment. And more important, there is not a single type of distribution: most of the IMs actually exhibit a Gaussian-like distribution, but other types can be encountered. This is an important observation because it means that methods implemented for outlier detection cannot rely on a given distribution but have to be able to handle various types of distribution.

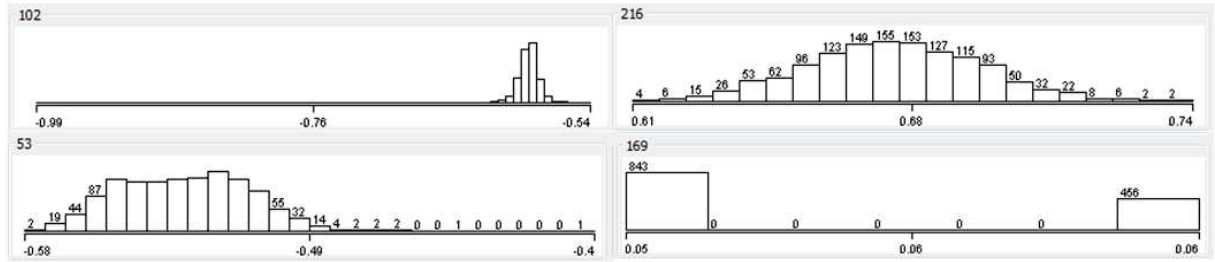


Figure 2.3: Examples of IM distribution encountered in the database (transceiver test vehicle)

2.1.2 Adaptive k-filter

Our proposal for screening out outliers from the database is to implement an adaptive filter that iteratively prunes the outliers. More specifically, the idea is to exploit the sensitivity of descriptive statistics to outliers and adapt the filter limits at each iteration. Details on this filter are given hereafter.

Let us consider the mean values $\mu = \{\mu_1, \dots, \mu_m\}$ and the standard deviations $\sigma = \{\sigma_1, \dots, \sigma_m\}$ of the m indirect measurements for a given dataset. As suggested in [48], a simple filter is a static filter, called k -filter, which is we can defined a filter as a hyper-rectangle in the indirect measurement space:

$$H_k = \{\mu_1 \pm k \cdot \sigma_1, \dots, \mu_m \pm k \cdot \sigma_m\} \quad (2.1)$$

where k is a positive real number. All devices whose indirect measurements fall outside the k -filter are considered as outliers and excluded from the dataset.

The main issue with this static k-filter is the choice of k . A strict k-filter (e.g. $k = 3$) will indeed exclude outliers, but it also excludes a number of circuits that actually belong to the tail of the distribution. However it is crucial to keep these tail devices during the learning phase so that the learned regression model covers the entire range of the distribution. Indeed during the testing phase, all devices whose indirect measurements fall outside the k-filter will be discarded since the model can be trusted only for devices whose indirect measurements fall inside the k-filter. Some of these devices actually correspond to failing devices affected by gross defect but other can be passing devices with indirect measurements just outside the k-filter. Simply rejecting all these devices therefore incurs unnecessary yield loss. Another option is to retest these devices through conventional measurements, which obviously incurs additional testing costs. With a strict k-filter, the learned model covers only a limited range of the distribution; the number of discarded devices during the testing phase will be high, therefore limiting the benefit alternate test. In contrast, with a lenient k-filter (e.g. $k > 10$), only few outliers will be identified and many circuits that do not belong to the distribution will remain in the dataset, affecting the accuracy of the model learned during the training phase.

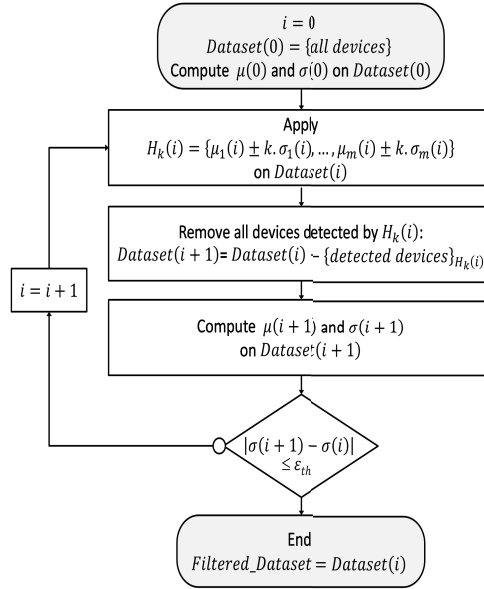
Our proposal to cope with this issue is to perform outlier detection based on iterative application of a moderate k-filter (typically $6 \leq k < 10$), with an adjustment of the k-filter limits at each iteration. In particular the idea is to recalculate, at each iteration, the mean and standard deviation of the dataset after exclusion of the devices detected by the k-filter. Since these statistics are sensitive to outliers, if the new values differ from the preceding ones, we can consider that excluded devices actually correspond to outliers and we go to the next iteration with updated k-filter limits. On the contrary if the mean and standard deviation are not affected, we can consider that the dataset no longer contains outliers but only devices that belong to the distribution; the procedure is then stopped. This procedure is summarized in figure 2.4.

The adaptive k-filter has been applied on the database of the two test vehicles, considering different values of k between 6 and 10. Results are summarized in Table 2.1 and 2.2 which report the number of circuits eliminated at each iteration for the PA and transceiver test vehicles respectively. For the PA test vehicle, the adaptive k-filter has almost no effect on

Table 2.1: Number of outliers eliminated from the PA database for different values of k

k	Iter1	Iter2	Iter3	Iter Total
6	3	2	0	5(0.045%)
7	1	1	0	2(0.017%)
8	1	1	0	2(0.017%)
9	0			0
10	0			0

the database: only less than 0.1% of the circuits are detected as outlier for k equal to 6 or 7, and no circuit for k superior or equal to 8. This is actually consistent with the fact that NXP has already applied an empirical filter on the database, considering "ad-hoc" defined test limits on each indirect measurements. The interest of the proposed filter is that detection of

**Figure 2.4:** Adaptive k-filter**Table 2.2:** Number of outliers eliminated from the transceiver database for different values of k

k	Iter1	Iter2	Iter3	Iter4	Iter5	Iter6	Iter7	Iter8	Iter9	Iter Total
6	116	33	14	4	1	1	0			169(13%)
7	76	30	11	2	2	1	0			122(9%)
8	51	20	3	2	1	1	1	0		78(6%)
9	46	9	3	2	1	1	1	1	0	64(5%)
10	43	7	1	1	1	1	1	1	0	56(4%)

outliers is performed automatically without any "ad-hoc" settings.

For the transceiver test vehicle, NXP has provided us the raw database without any empirical filtering. In this case, the adaptive k-filter permits to discard a number of circuits from the database. As expected the smaller the value of k , the higher the number of circuits identified as outliers: 4% of the circuits are identified as outliers with $k = 10$, and 13% for $k = 6$. The choice of $k = 6$ seems an appropriate option that preserves the large majority of data but effectively excludes circuits that do not belong to the global statistical distribution. Figure 2.5 illustrates the operation of the filter with $k = 6$ for some examples of IM distribution encountered in the database. These examples show that the filter correctly identifies outliers (when present) while maintaining the global IM statistical characteristics, whatever the type of IM distribution. The benefit from the filter application is illustrated in figure 2.6, which

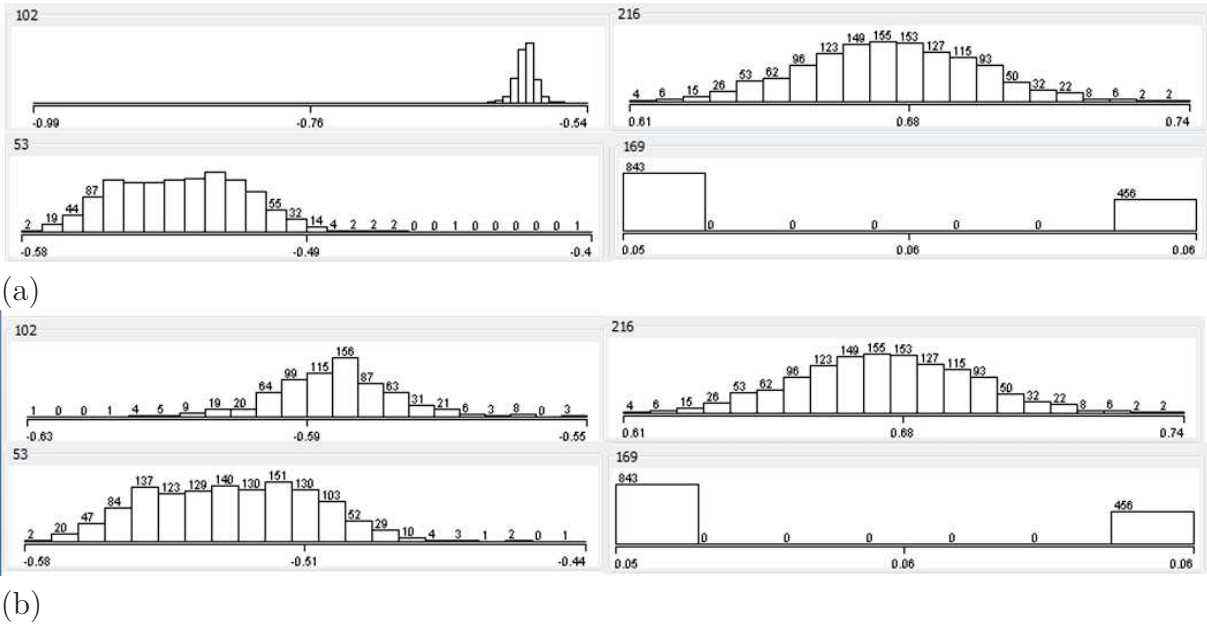


Figure 2.5: Examples of IM distribution before (a) and after (b) the filtering process (transceiver test vehicle)

compares the correlation coefficient (R) and the estimation errors (ε_{rms-TS} and ε_{max-TS}) achieved for a model built on the initial database and for the same model built on the filtered database. These results clearly show that the presence of outliers indeed affects the regression model quality and demonstrate the effectiveness of the filter: by removing these outliers, there is an increase of the correlation coefficient and a significant improvement of both the average and maximal estimation errors.

To conclude this section, table 2.3 summarizes the database composition for both test vehicles after application of the adaptive k-filter. All experiments and results presented in the remaining of the manuscript will be conducted on these filtered databases.

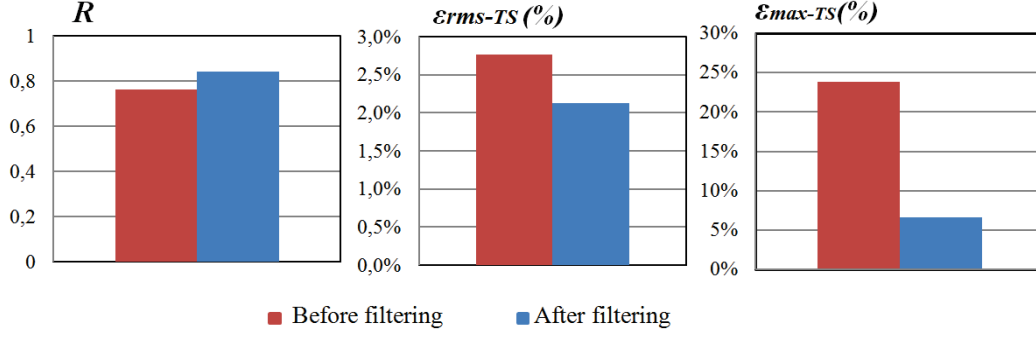


Figure 2.6: Correlation coefficient and estimation errors before and after the filtering process (transceiver test vehicle)

Table 2.3: Test vehicle databases after outlier filtering

Test vehicle	Number of circuits	Number of IMs	Number of measured Performances (Ps)
PA	11202	37	1
Transceiver	1130	405	1

2.2 Methods for indirect measurement selection

An essential step for efficient implementation of the alternate test strategy is the choice of adequate indirect measurements for each circuit performance to be predicted, which have to be information-rich and correlate well with the device performances. However this is a circuit-specific problem and there is no proven generic method to identify such indirect measurements. Very often, indirect measurements are defined ad-hoc, based on the expertise of the designer and the precise knowledge of the DUT. To illustrate how important is this choice, figure 2.7 shows an example of scatter plot that can be obtained for the transceiver test vehicle with a random choice of three indirect measurements. Obviously in this case, the IMs used to build the model are completely irrelevant since there is absolutely no correlation between predicted and actual values of the device performance.

Note that the choice of adequate indirect measurements is not an easy task because there are many possibilities for obtaining indirect measurements:

- standard external DC tests typically performed on any device can be used as indirect measurements [9]
- the device may be equipped with DC probes and a DC test bus allowing measurements on internal nodes [14],
- the device may also include built-in sensors such as envelope sensors, dummy circuits, PCM which offer additional indirect measurements [49][50],
- finally all these measurements can be performed under different test conditions, for example multi-vdd conditions [51], which allows to multiply the number of indirect measurements

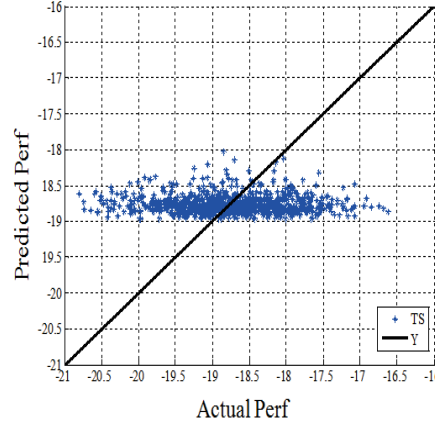


Figure 2.7: Correlation graph for a transceiver performance performed on training set (TS)

by the number of different settings.

Then the question is: why not building regression models that use all possible indirect measurement candidates? There are actually several reasons for this. First the computational effort for building a model depends on the number of input variables used in the model and therefore by using only a limited number of indirect measurements, the computational burden is reduced. Second by using only a limited number of indirect measurements, the number of actual measurements that have to be performed during production test is limited, therefore reducing the test cost. But the essential reason for using only a limited number of indirect measurements is to avoid the curse of dimensionality, which is a critical issue for the accuracy of test predictions. Indeed in the context of regression analysis, curse of dimensionality means that the predictive power of a model built with a given number of training samples reduces as the dimensionality increases. This is a well-identified phenomenon, as illustrated on the graph of figure 2.8 coming from a tutorial on "Machine Learning and its Applications in Test" [52]: the test error computed on training devices continuously decreases by adding new indirect measurements to the model, while the test error computed for new devices first decreases but then increases as the prediction model uses more and more indirect measurements. This is therefore the fundamental reason for using only a limited number of indirect measurements.

To summarize, on the one hand it's easy to identify a large set of indirect measurements candidates, but on the other hand it's mandatory to use only a limited number of indirect measurement candidates. The problem is therefore how to select a pertinent subset of indirect measurements from a large set of IM candidates. This is precisely the role of feature selection, also known as variable selection, attribute selection or variable subset selection, which is the process of selecting one subset of relevant features (variables, predictors) for use in model construction. Note that feature selection algorithms are classically directed towards the selection of a single subset; they can be used in the context of classical indirect test implementation, where a single prediction model is required for each device performance.

Feature selection algorithms can actually be classified in two main categories: filter methods and wrapper methods as illustrated in figure 2.9 [53]. Filter methods perform feature selection

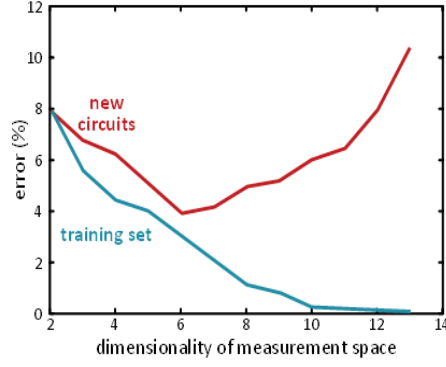


Figure 2.8: Curse of dimensionality issue in the context of indirect testing

based on ranking the variables according to a given statistical observations. It is therefore a pre-processing step, independent of the machine-learning prediction model. In contrast, the wrapper methodology consists in using the prediction performance of a given machine-learning prediction model to assess the relative usefulness of subsets of variables. This information is used within an optimization loop in order to guide the search for efficient subsets of variables.

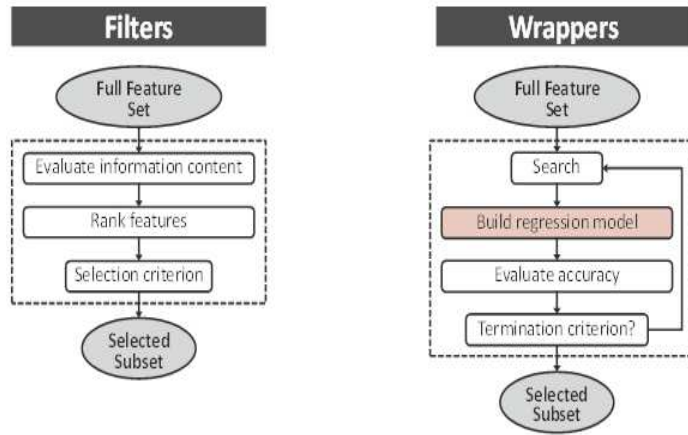


Figure 2.9: The two main categories of feature selection algorithms

In this section, we present four different strategies of indirect measurement selection, pertaining to both categories. These strategies will be applied to the two test vehicles and a comparative analysis presented in the following sections.

2.2.1 IM selection based on Pearson correlation

A first simple strategy to perform IM selection is based on the computation of sample Pearson correlation coefficient, which is a statistical measure of the linear dependence between two variables. Pearson correlation coefficient is actually defined as the covariance of the two variables divided by the product of their standard deviations:

$$R(X, Y) = \frac{cov(X, Y)}{\sigma(X)\sigma(Y)} \quad (2.2)$$

where cov is the covariance and σ the standard deviation.

Considering (X_i, Y_i) , $i = 1, 2, \dots, n$ a statistical sample from a pair of random variables (X, Y) , the Pearson correlation coefficient can be expressed as:

$$R(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}} \quad (2.3)$$

where \bar{X} and \bar{Y} are the mean values of the variables X and Y respectively:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i, \text{ and } \bar{Y} = \frac{1}{n} \sum_{i=1}^n Y_i$$

In the context of IM selection strategy, Pearson correlation coefficient can be used to perform variable ranking. More precisely for a given performance P_j to be evaluated, indirect measurements IM_k are sorted in decreasing order of their Pearson correlation coefficient to the targeted performance $R(IM_k, P_j)$. IM selection is then performed by selecting only the most relevant variables, i.e. the highest-ranked indirect measurements. This strategy clearly belongs to the filter category, since it is a simple pre-processing step independent of the choice of any machine-learning prediction model.

2.2.2 IM selection based on Brownian distance correlation

Another filter method for IM selection has been recently introduced based on the use of the Brownian distance correlation instead of Pearson correlation coefficient [54]. In statistics and in probability theory, distance correlation is a measure of statistical dependence between two random variables or two random vectors of arbitrary, not necessarily equal dimension. This measure is derived from a number of other quantities that are used in its specification, specifically: distance variance, distance standard deviation and distance covariance. It has been introduced by G.J. Szekely to palliate a limitation of Pearson correlation, which assumes linear dependence conditions. Definitions are recalled here for completeness, more details can be found in [55]. Let us consider $(X_i, Y_i), i = 1, 2, \dots, n$ a statistical sample from a pair of real valued or vector valued random variables (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 $B_{j,i}$:

$$dCov_n^2(X, Y) = \frac{1}{n^2} \sum_{j,i=1}^n A_{j,i} B_{j,i} \quad (2.4)$$

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

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

where

$$\begin{aligned}\bar{a}_{j.} &= \frac{1}{n} \sum_{i=1}^n a_{j,i}, \bar{a}_{.i} = \frac{1}{n} \sum_{j=1}^n a_{j,i}, \bar{a}_{..} = \frac{1}{n^2} \sum_{j,i=1}^n a_{j,i} \\ \bar{b}_{j.} &= \frac{1}{n} \sum_{i=1}^n b_{j,i}, \bar{b}_{.i} = \frac{1}{n} \sum_{j=1}^n b_{j,i}, \bar{b}_{..} = \frac{1}{n^2} \sum_{j,i=1}^n b_{j,i}\end{aligned}$$

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:

$$dVar_n^2(X) = dCov_n^2(X, X) = \frac{1}{n^2} \sum_{i=1}^n A_{j,i}^2 \quad (2.5)$$

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):

$$dCor_n(X, Y) = \frac{dCov_n(X, Y)}{\sqrt{dVar_n(X) dVar_n(Y)}} \quad (2.6)$$

In the context of IM selection strategy, the idea is to use distance correlation between indirect measurements and targeted performances in order to perform a priori selection, i.e. before training any machine-learning model. In this strategy, for a given performance P_j to be evaluated, indirect measurements IM_k are sorted in decreasing order of their distance correlation to the targeted performance $dCor_n(IM_k, P_j)$. Only the most relevant indirect measurements, i.e. the highest-ranked indirect measurements, are then selected to build the regression model.

2.2.3 IM selection based on SFS algorithm

A different strategy is to include the machine-learning model in the selection algorithm in order to score subsets of variables according to their prediction performance. In practice, the following points have to be specified to define a wrapper method: (i) how to search the space of all possible variable subsets; (ii) how to assess the prediction performance of a learning machine to guide the search and halt it; and (iii) which learning-machine model to use.

An exhaustive search can conceivably be performed, if the number of variables is not too large. However the problem is known to be NP-hard and the search becomes quickly computationally intractable as dimensionality rises. Indeed for a given performance P_j to be evaluated, the number of models M that needs to be evaluated in case of a full exploration of all possible subsets composed of up to k IMs is given by:

$$M = \sum_{i=1}^k C_m^i \quad (2.7)$$

where m is the number of IM candidates, k the maximum number of selected IMs, and C_m^i refers to the binomial coefficient (number of combinations of i IMs over m).

As a numerical example, the exploration of all possible subsets composed of up to 10 IMs

selected among 40 candidates necessitates the building of more than $1.22e9$ models. Assuming a processing time of 0.1s to build and evaluate one model, this would necessitate more than 3.87 years which is clearly impractical. Other search strategies have therefore to be implemented to handle this computational issue.

A wide range of search strategy can be used, including best-first, branch-and-bound, simulated annealing, genetic algorithm. One of the simplest strategies is the best-first strategy, in which the best candidate is added to the solution at each iteration. Figure 2.10 depicts the simplified diagram of such an iterative search, also known as Sequential Forward Selection (SFS), for a given performance P_j , as suggested in [50] [56].

In this case, MARS is chosen as the learning-machine prediction model and performance prediction is evaluated using the rms prediction error ε_{rms-TS} expressed as percentage, computed by the Normalized Root Mean Square Error (NRMSE) on training devices by equation (1.4).

The first iteration of the search algorithm consists in building a MARS model for each IM_k and selecting the one that generates the model with the minimum rms prediction error ε_{rms-TS} . In the second iteration, a MARS model is built for each pair (IM_k, IM_l) including previously selected one, and the pair that gives the model with minimum rms prediction error ε_{rms-TS} is selected. Then, we work with triplets, always keeping the IMs selected in the previous iterations and so on. The procedure stops when the obtained rms prediction error is below a pre-defined target value $\varepsilon_{rms-TS} \leq \varepsilon_{target}$ or when the maximum number of selected IMs is reached.

This procedure is very efficient in terms of processing time. Indeed the number of models to

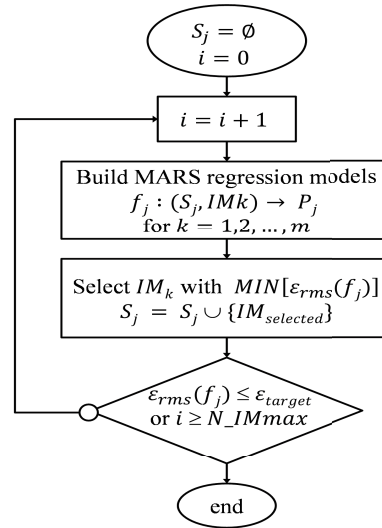


Figure 2.10: SFS search of an IM subset for prediction of one specification P_j

be built with the SFS strategy in order to select a subset of up to p IMs among a set of m candidates is given by:

$$M_SFS = \sum_{i=1}^p C_{m-k+1}^1 = p * (m + 1) - (p * (p + 1))/2 \quad (2.8)$$

This corresponds to the construction of only 355 models for the case study already mentioned above (selection of a subset of up to 10 IMs among 40 candidates).

2.2.4 IM selection using MARS built-in selection feature

Finally the last investigated strategy is to use the built-in selection feature of the Multivariate Adaptive Regression Splines (MARS) algorithm as suggested in [57], which also belongs to the wrapper category. A user-configurable limit of maximum allowed basis functions(NBFmax) can be controlled to fix the number of IMs selected by this latter.

2.3 Experimental setup for the evaluation of IM selection methods

In order to perform a comparative analysis of the different strategies for indirect measurement selection, the experimental setup described in Figure 2.11 was developed. The objective is to evaluate the test efficiency achieved by the alternate test procedure when using the indirect measurements selected by a given algorithm. The experimental setup therefore involves two distinct steps, namely (i) IM selection and (ii) indirect test efficiency evaluation. Details on each step are given in the following subsections. Note that to perform this evaluation, the full dataset of devices that contains all available measurements (including both the regular specification measurements and the low-cost indirect measurements) is separated into two distinct sets, i.e. one dataset of training devices and one dataset of validation devices. For a realistic evaluation, the validation set should be larger than the training set since the objective of the alternate test strategy is actually to predict the performances of a large number of devices during mass production testing using only a limited number of devices measured in the training phase. In this work, a Latin Hyper-Cube Sampling process (LHCS) is used to implement this separation such that both datasets exhibit similar statistical characteristics. The dataset of training devices is used both in the IM selection and evaluation steps, while the dataset of validation devices is only used in the evaluation step.

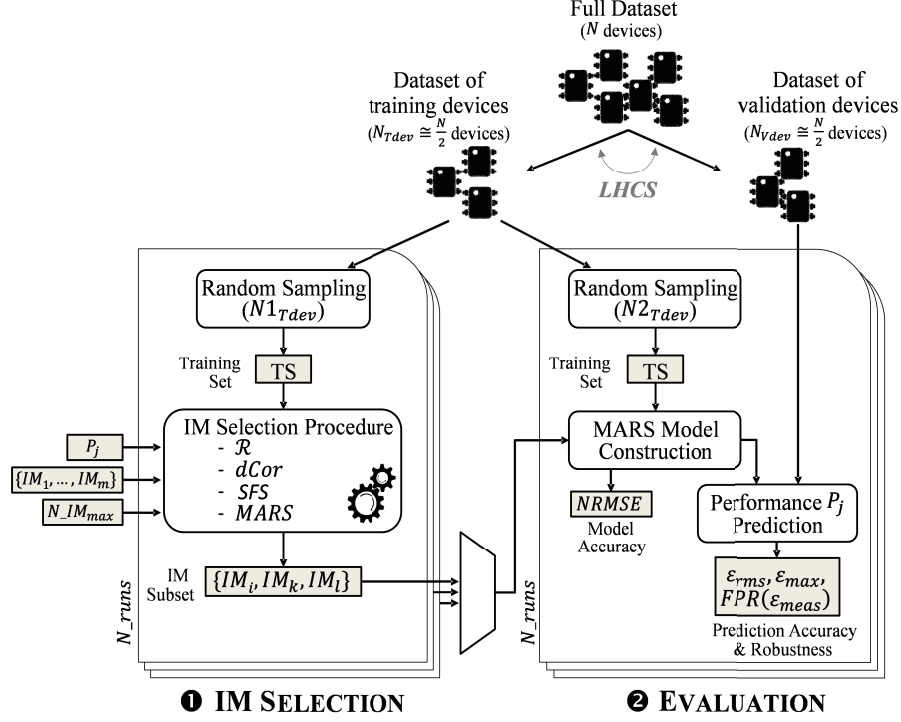


Figure 2.11: Experimental setup for test efficiency evaluation

2.3.1 IMs selection

The purpose of this step is to select an IM subset of a given size N_IM_{max} that will be used to predict a given analog/RF performance P_j from a large set of indirect measurement candidates $\{IM_1, IM_2, \dots, IM_m\}$. In this step, the four different strategies presented in the previous section are implemented.

The influence of the training set is also investigated. Indeed some studies have been performed regarding the influence of the training set (in particular the size of the training set) on the prediction accuracy [58], but these studies rely on given pre-selected indirect measurements to perform the prediction. We have found no study in the literature regarding the influence of the training set on the selection of indirect measurements and the resulting impact on the prediction accuracy. It is therefore also the objective of this chapter to analyze this aspect.

For this, we consider training sets of different size (composed of $N1_{Tdev}$ devices) obtained with a random sampling of the dataset of training devices. In addition for a given subset size, the variability of the training set is also investigated by considering a number of different random sampling runs (N_{runs}). The selected subset of IMs then corresponds to the most frequently chosen subset (according to the used IM selection algorithm) over the different random runs. Figure 2.12 below shows as an example for the transceiver test vehicle, the histogram of subsets composed of 3 IMs selected by the MARS built-in selection feature over 100 random runs. The IM subset IM_{24}, IM_6, IM_{36} is the most frequently chosen subset and will be selected for the evaluation step..

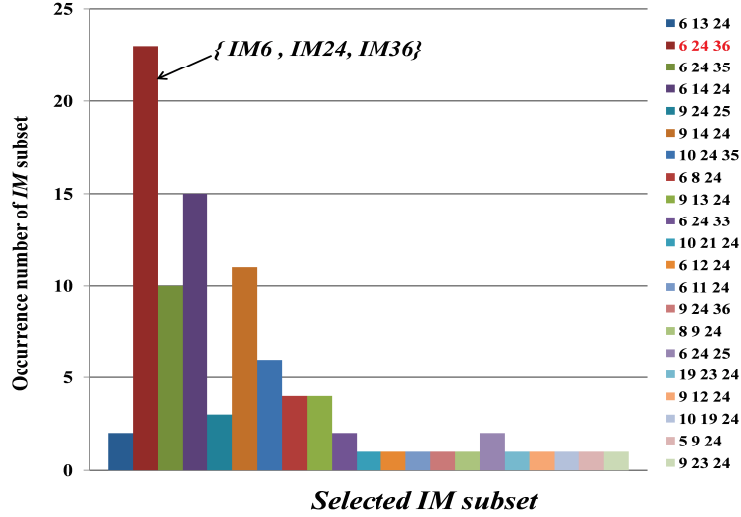


Figure 2.12: Illustration of chosen IM subsets over 100 runs (transceiver test vehicle)

2.3.2 Indirect test efficiency evaluation

The purpose of this step is to evaluate the efficiency of indirect test implementations based on the IM subsets selected in the previous step, in terms of accuracy and robustness of the predictions achieved on the dataset of validation devices.

In this step, IM subsets selected in the previous step are used to build regression models using MARS algorithm. Models are constructed for different test cases corresponding to a given performance P_j to be predicted, a given strategy to perform IM selection, a given value N_IM_{max} for the maximum number of IMs used to predict the performance, and a given training set. Note that here again the influence of the training set size and composition is investigated by selecting training sets of different sizes with a random sampling of the dataset of training devices (N_{2Tdev} devices) and considering a number of different random sampling runs (N_{runs}) for each training set size.

More precisely, for each test case, measurement data of the targeted performance together with measurement data of the corresponding selected IMs are provided as inputs to the MARS algorithm. The predictive model is then constructed as a linear combination of basis functions, which involve only the selected IMs. Note that in this case, the built-in selection feature of the algorithm is not used.

The model is then used to perform performance prediction over the dataset of validation devices and test efficiency is evaluated in terms of both prediction accuracy and prediction reliability. More particularly, we compute the classical accuracy metrics (rms and maximal prediction errors defined in the first chapter by equations (1.4) and (1.5)), and the new proposed reliability metric (Failing Prediction Rate, FPR, defined in the first chapter by equation (1.6)).

Finally note that the same validation devices are always used over the different test cases in order to allow consistent comparison.

2.4 Results and discussion

Experiments were performed considering the Power Amplifier (PA) and the RF transceiver test vehicles. The typical measurement repeatability error for those test vehicles performances are $\varepsilon_{meas} = 3\%$ and $\varepsilon_{meas} = 5\%$ respectively.

For the two test vehicles, the full dataset of N devices is first separated in two sets of equivalent size using a LHCS process, one dataset containing N_{Tdev} training devices and one dataset containing N_{Vdev} validation devices. Then a campaign of experiments is conducted applying the experimental setup of Figure 2.11 with the following parameters:

- Number of IMs to be selected: $N_{IM_{max}} = \{1, 2, 3, 4, 5\}$
- Random sampling of Training Set for IM selection:

- ◇ 50 runs with $N1_{Tdev} = 80\% * N_{Tdev}$.
- ◇ 100 runs with $N1_{Tdev} = 40\% * N_{Tdev}$.
- ◇ 200 runs with $N1_{Tdev} = 20\% * N_{Tdev}$.

- Random sampling of Training Set for test efficiency evaluation:

- ◇ 50 runs with $N2_{Tdev} = 80\% * N_{Tdev}$.
- ◇ 100 runs with $N2_{Tdev} = 40\% * N_{Tdev}$.
- ◇ 200 runs with $N2_{Tdev} = 20\% * N_{Tdev}$.

2.4.1 IM selection

Results of the IM selection procedure for the two test vehicles are summarized in tables 2.4 and 2.5 respectively.

Table 2.4: Selected IM subsets for the PA test vehicle according to different selection strategies, different training set sizes, and different values for the maximum number of IMs used to predict the performance

	$N1_{dep}$	Filter Methods		Wrapper Methods	
		Pearson-Corr Ranking	Distance-Corr Ranking	SFS Search	MARS Built-in Selection
$N_{IM_{max}} = 1$	1,000	$\{IM_{14}\}$	$\{IM_{24}\}$	$\{IM_{24}\}$	$\{IM_{24}\}$
	2,000	$\{IM_{14}\}$	$\{IM_{24}\}$	$\{IM_{24}\}$	$\{IM_{24}\}$
	4,000	$\{IM_{14}\}$	$\{IM_{24}\}$	$\{IM_{24}\}$	$\{IM_{24}\}$
$N_{IM_{max}} = 2$ 40	1,000	$\{IM_{14}, IM_{15}\}$	$\{IM_{24}, IM_9\}$	$\{IM_{24}, IM_6\}$	$\{IM_{24}, IM_{13}\}$
	2,000	$\{IM_{14}, IM_{15}\}$	$\{IM_{24}, IM_9\}$	$\{IM_{24}, IM_9\}$	$\{IM_{24}, IM_9\}$
	4,000	$\{IM_{14}, IM_{15}\}$	$\{IM_{24}, IM_9\}$	$\{IM_{24}, IM_{21}\}$	$\{IM_{24}, IM_6\}$
$N_{IM_{max}} = 3$	1,000	$\{IM_{14}, IM_{15}, IM_{12}\}$	$\{IM_{24}, IM_9, IM_{13}\}$	$\{IM_{24}, IM_6, IM_{10}\}$	$\{IM_{24}, IM_{13}, IM_6\}$
	2,000	$\{IM_{14}, IM_{15}, IM_{12}\}$	$\{IM_{24}, IM_9, IM_{13}\}$	$\{IM_{24}, IM_9, IM_{10}\}$	$\{IM_{24}, IM_9, IM_{14}\}$
	4,000	$\{IM_{14}, IM_{15}, IM_{12}\}$	$\{IM_{24}, IM_9, IM_{13}\}$	$\{IM_{24}, IM_{21}, IM_{37}\}$	$\{IM_{24}, IM_6, IM_{36}\}$
$N_{IM_{max}} = 4$	1,000	$\{IM_{14}, IM_{15}, IM_{12}, IM_{11}\}$	$\{IM_{24}, IM_9, IM_{13}, IM_{15}\}$	$\{IM_{24}, IM_6, IM_{10}, IM_{37}\}$	$\{IM_{24}, IM_{13}, IM_6, IM_{33}\}$
	2,000	$\{IM_{14}, IM_{15}, IM_{12}, IM_{11}\}$	$\{IM_{24}, IM_9, IM_{13}, IM_{15}\}$	$\{IM_{24}, IM_9, IM_{10}, IM_{15}\}$	$\{IM_{24}, IM_6, IM_{25}, IM_8\}$
	4,000	$\{IM_{14}, IM_{15}, IM_{12}, IM_{11}\}$	$\{IM_{24}, IM_9, IM_{13}, IM_{15}\}$	$\{IM_{24}, IM_{21}, IM_{37}, IM_6\}$	$\{IM_{24}, IM_6, IM_{25}, IM_{33}\}$
$N_{IM_{max}} = 5$	1,000	$\{IM_{14}, IM_{15}, IM_{12}, IM_{11}, IM_{10}\}$	$\{IM_{24}, IM_9, IM_{13}, IM_{15}, IM_{11}\}$	$\{IM_{24}, IM_6, IM_{10}, IM_{37}, IM_{25}\}$	$\{IM_{24}, IM_{13}, IM_6, IM_{33}, IM_{37}\}$
	2,000	$\{IM_{14}, IM_{15}, IM_{12}, IM_{11}, IM_{10}\}$	$\{IM_{24}, IM_9, IM_{13}, IM_{15}, IM_{11}\}$	$\{IM_{24}, IM_9, IM_{10}, IM_{15}, IM_{16}\}$	$\{IM_{24}, IM_6, IM_{25}, IM_8, IM_{36}\}$
	4,000	$\{IM_{14}, IM_{15}, IM_{12}, IM_{11}, IM_{10}\}$	$\{IM_{24}, IM_9, IM_{13}, IM_{15}, IM_{11}\}$	$\{IM_{24}, IM_{21}, IM_{37}, IM_6, IM_{16}\}$	$\{IM_{24}, IM_6, IM_{25}, IM_{33}, IM_{36}\}$

Table 2.5: Selected IM subsets for the PA test vehicle according to different selection strategies, different training set sizes, and different values for the maximum number of IMs used to predict the performance

	$N1_{Tdev}$	Filter Methods		Wrapper Methods	
		Pearson-Corr Ranking	Distance-Corr Ranking	SFS Search	MARS Built-in Selection
$N_{IM_{max}} = 1$	1,000	$\{IM_{219}\}$	$\{IM_{219}\}$	$\{IM_{219}\}$	$\{IM_{219}\}$
	2,000	$\{IM_{219}\}$	$\{IM_{219}\}$	$\{IM_{219}\}$	$\{IM_{219}\}$
	4,000	$\{IM_{219}\}$	$\{IM_{219}\}$	$\{IM_{219}\}$	$\{IM_{219}\}$
$N_{IM_{max}} = 2$ 42	1,000	$\{IM_{219}, IM_{218}\}$	$\{IM_{219}, IM_{218}\}$	$\{IM_{219}, IM_{213}\}$	$\{IM_{219}, IM_{213}\}$
	2,000	$\{IM_{219}, IM_{218}\}$	$\{IM_{219}, IM_{218}\}$	$\{IM_{219}, IM_{206}\}$	$\{IM_{219}, IM_{213}\}$
	4,000	$\{IM_{219}, IM_{218}\}$	$\{IM_{219}, IM_{218}\}$	$\{IM_{219}, IM_{212}\}$	$\{IM_{219}, IM_{213}\}$
$N_{IM_{max}} = 3$	1,000	$\{IM_{219}, IM_{218}, IM_{217}\}$	$\{IM_{219}, IM_{218}, IM_{217}\}$	$\{IM_{219}, IM_{213}, IM_{331}\}$	$\{IM_{219}, IM_{213}, IM_{17}\}$
	2,000	$\{IM_{219}, IM_{218}, IM_{217}\}$	$\{IM_{219}, IM_{218}, IM_{217}\}$	$\{IM_{219}, IM_{206}, IM_{161}\}$	$\{IM_{213}, IM_{219}, IM_{196}\}$
	4,000	$\{IM_{219}, IM_{218}, IM_{217}\}$	$\{IM_{219}, IM_{218}, IM_{217}\}$	$\{IM_{219}, IM_{212}, IM_{232}\}$	$\{IM_{219}, IM_{213}, IM_{196}\}$
$N_{IM_{max}} = 4$	1,000	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}\}$	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}\}$	$\{IM_{219}, IM_{213}, IM_{313}, IM_{196}\}$	$\{IM_{219}, IM_{213}, IM_{212}, IM_{215}\}$
	2,000	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}\}$	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}\}$	$\{IM_{219}, IM_{206}, IM_{161}, IM_{196}\}$	$\{IM_{219}, IM_{213}, IM_{292}, IM_{209}\}$
	4,000	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}\}$	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}\}$	$\{IM_{219}, IM_{212}, IM_{232}, IM_{196}\}$	$\{IM_{212}, IM_{213}, IM_{196}, IM_{169}\}$
$N_{IM_{max}} = 5$	1,000	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}, IM_{215}\}$	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}, IM_{215}\}$	$\{IM_{219}, IM_{213}, IM_{331}, IM_{196}, IM_{204}\}$	$\{IM_{219}, IM_{213}, IM_{212}, IM_{215}, IM_{223}\}$
	2,000	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}, IM_{215}\}$	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}, IM_{215}\}$	$\{IM_{219}, IM_{206}, IM_{161}, IM_{196}, IM_{204}\}$	$\{IM_{219}, IM_{292}, IM_{209}, IM_{314}, IM_{323}\}$
	4,000	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}, IM_{215}\}$	$\{IM_{219}, IM_{218}, IM_{217}, IM_{216}, IM_{215}\}$	$\{IM_{219}, IM_{212}, IM_{232}, IM_{196}, IM_{204}\}$	$\{IM_{219}, IM_{213}, IM_{196}, IM_{169}, IM_{331}\}$

A first observation is that for both test vehicles, the selected subset does not depend on the Training Set Size (TSS) when using filter methods while it is sensitive to this parameter when using wrapper methods, in particular when increasing the number of selected IMs. A second observation is that the two filter methods produce identical results in case of the transceiver test vehicle while selected subsets differ in case of the PA test vehicle. Regarding wrapper methods, they actually lead to different selected subsets, even if there are some common points. For instance for the PA test vehicle, the indirect measurement IM_{24} is present in every subset selected either by the SFS search or MARS built-in feature. In the same way for the transceiver test vehicle, the indirect measurement IM_{219} is present in every selected subset, whatever the selection strategy.

In terms of computational time, experiments have revealed that the computational time required for the selection of one subset (one random sampling run) increases with the size of the training set. However it remains in the order of few seconds when using Pearson correlation ranking or MARS built-in feature while it is in the order of several tens of minutes when using the SFS search and several hours when using distance correlation ranking. Table 2.6 below illustrates the computational time for the transceiver test vehicle for different training set sizes (TSS).

Table 2.6: Computational time for the transceiver test vehicle

	SFS	MARS	Dist-Corr
$TSS = 80\% * N_{Tdev}$	$\sim 16minutes$	$\sim 3seconds$	$\sim 5hours$
$TSS = 40\% * N_{Tdev}$	$\sim 32minutes$	$\sim 8seconds$	$\sim 30hours$
$TSS = 20\% * N_{Tdev}$	$\sim 50minutes$	$\sim 21seconds$	$\sim 3days$

2.4.2 Test efficiency

Regarding test efficiency evaluation, the first step consists in building models for each one of the selected IM subsets, considering training set of different sizes. The intrinsic accuracy of the resulting models, i.e. their ability to accurately represent the devices used during the learning, is evaluated in terms of Normalized Root Mean Square Error (NRMSE) computed on the learning devices (mean value over the different sampling runs performed for each training set size). Figures 2.13 and 2.14 report this accuracy according to the number of selected IMs with the different strategies, for the PA and transceiver test vehicles respectively. Note that the same legend convention will be used for figures presented in the rest of chapter.

For both test vehicles, it can be observed that the size of the training set used to build the model ($N2_{Tdev}$) has no significant influence on the model accuracy. Also it can be observed that, although the selection procedure is sensitive to the training set size ($N1_{Tdev}$) when using wrapper methods since selected IM subsets differs for training sets of different size, there is no real impact in terms of model accuracy. For the PA test vehicle, the selection strategy based on Pearson correlation ranking leads to inferior performances compared to the three other strategies, especially when only few IMs are selected. More specifically, the NRMSE decreases from 1.65% down to about 0.8% when increasing the number of selected IMs in case

of selection based on Pearson correlation, and from 1% down to about 0.75% for the three other strategies.

For the transceiver test vehicle, we observe a different behavior depending whether IM selection is performed by filter or wrapper methods, with better performances when using wrapper methods. Indeed in this case we observe a constant NRMSE of about 2% for models built with at least 2 IMs, while it is necessary to use 5 IMs to reach almost a similar performance when using IM subsets selected by filter methods.

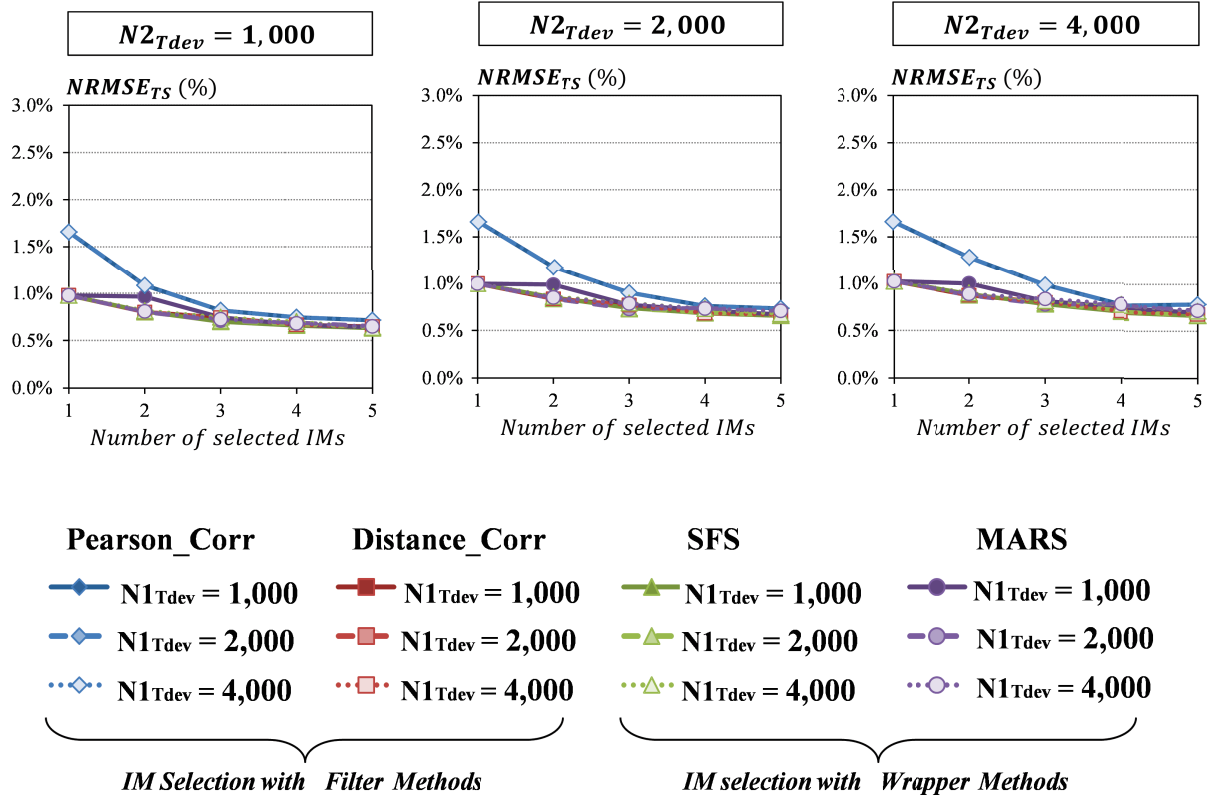


Figure 2.13: Model accuracy for the PA test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})

The following step of the evaluation procedure consists in using the models to perform performance prediction for all devices of the validation set. Results are summarized in figures 2.15 and figures 2.16 for the PA and transceiver test vehicles respectively, which report both the rms prediction error (mean value over the different sampling runs performed for each training set size) and maximal prediction error (worst case over the different sampling runs performed for each training set size) according to the number of selected IMs.

For the PA test vehicle, it appears that prediction accuracy achieved by the different models is influenced by many factors. In particular, it can be observed that the rms prediction error varies between 0.8% up to 2% depending on the selection strategy, the number of selected IMs, the size of the training set used during selection procedure and the size of the training set used to build the model. There is no evident trend, but two interesting points can be outlined. First, the lowest rms prediction error is achieved only for the largest size of training set

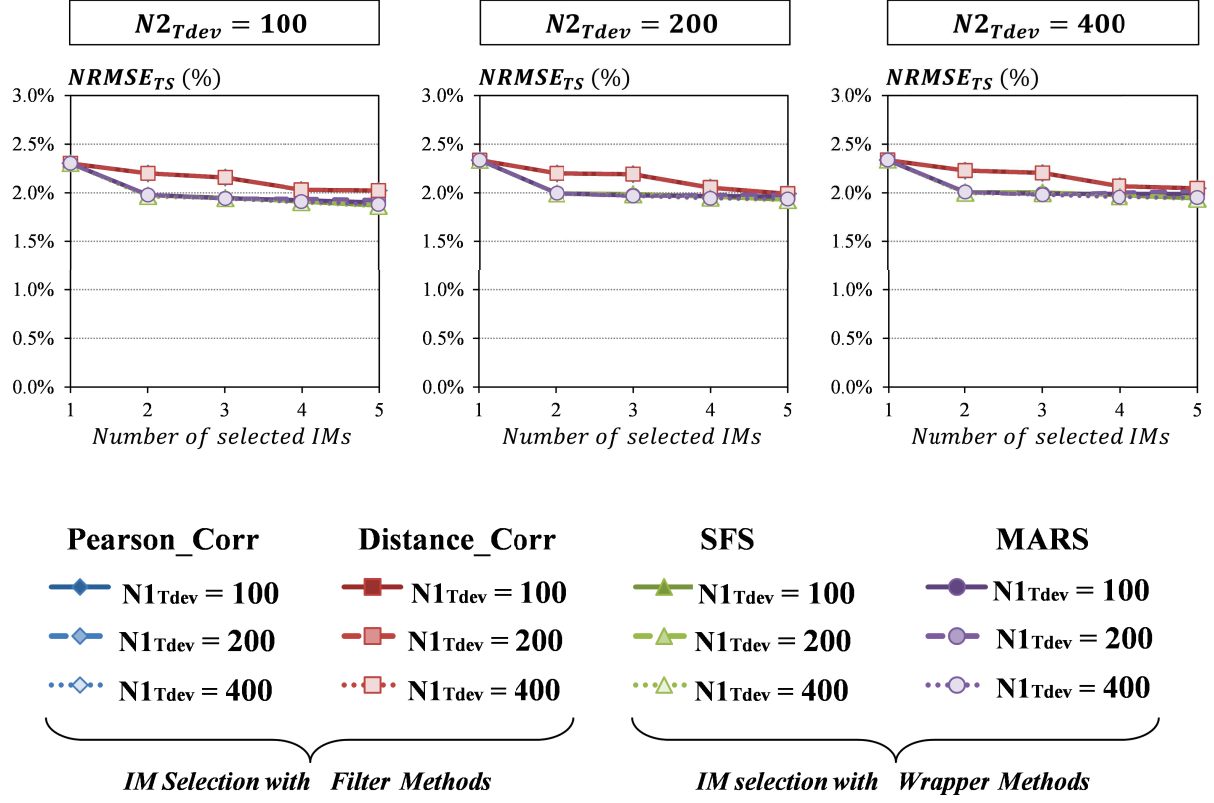
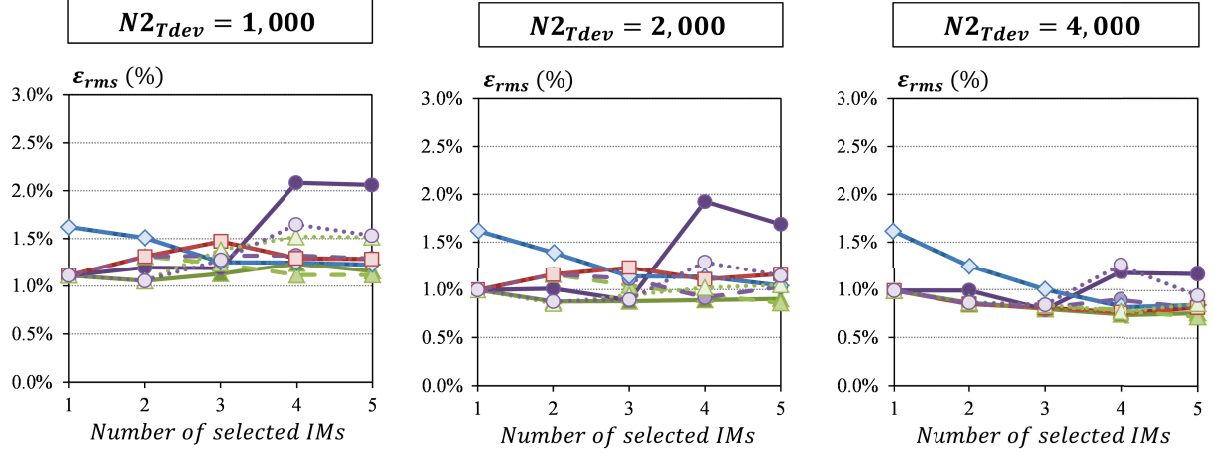
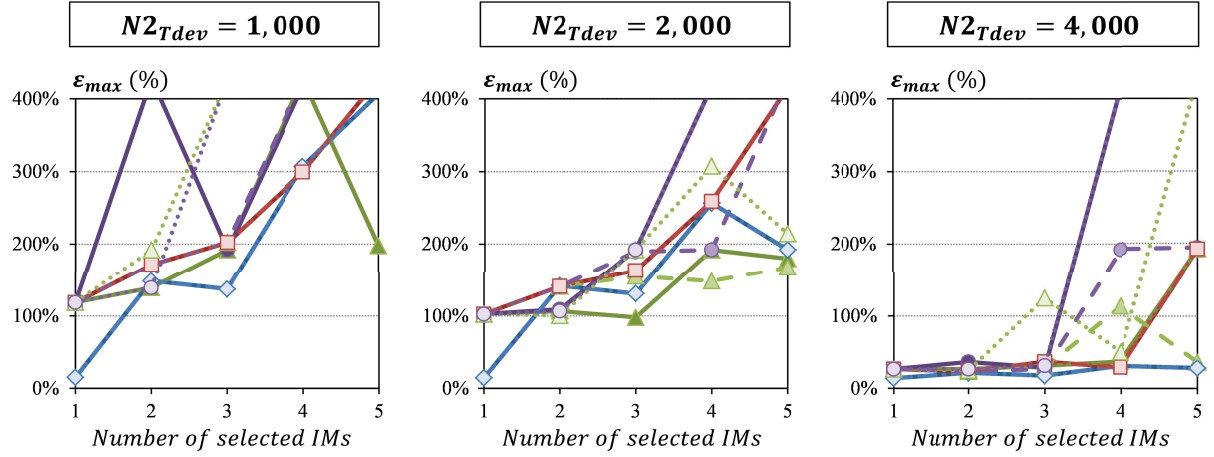


Figure 2.14: Model accuracy for the transceiver test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})

(a) rms prediction error ε_{rms} (mean value over N_{runs})(b) Maximal prediction error ε_{max} (worst case value over N_{runs})**Figure 2.15:** Prediction accuracy for the PA test vehicle considering different IM selection strategies and different sizes of training set

used to build the model. Second, increasing the number of selected IMs does not necessarily improve prediction accuracy, especially for models built using IM subsets selected by MARS built-in feature with a rms prediction error higher for models built with 4 or 5 IMs than for models built with fewer IMs. Regarding maximal prediction error, results exhibit a very large dispersion with a maximal prediction error that can vary between 25% up to more than 400%. Yet similar observations than for rms prediction error can be made, i.e. best results are obtained when models are built on a training set of 4,000 devices, and models built with 4 or 5 IMs can produce worse maximal error than models built with fewer IMs. However it is worth noting that even in the best case, the maximal prediction error remains extremely high for this test vehicle, i.e. higher than 20% which corresponds to about 25 times the rms prediction error.

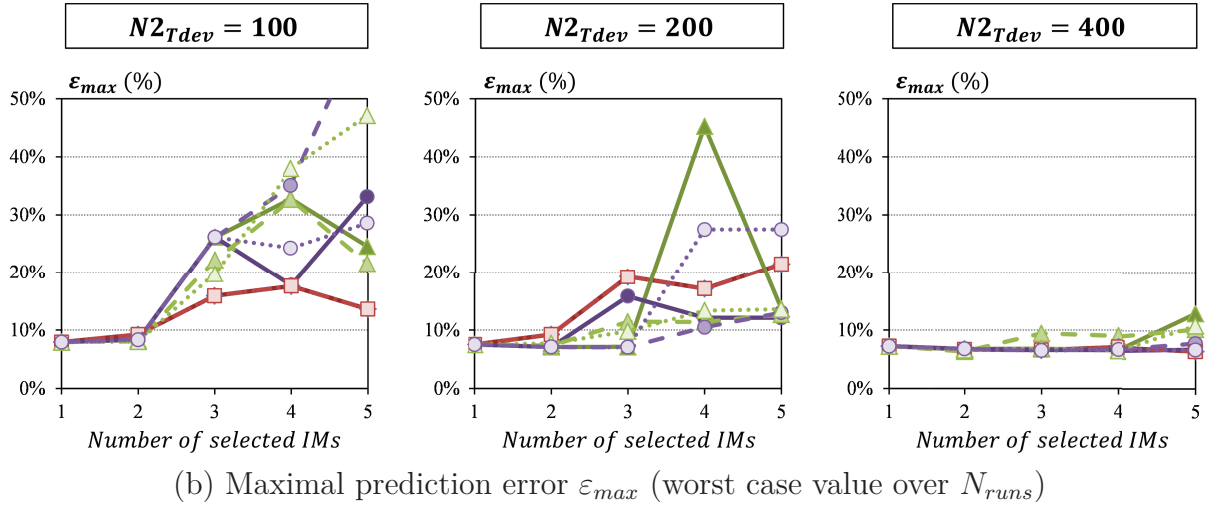
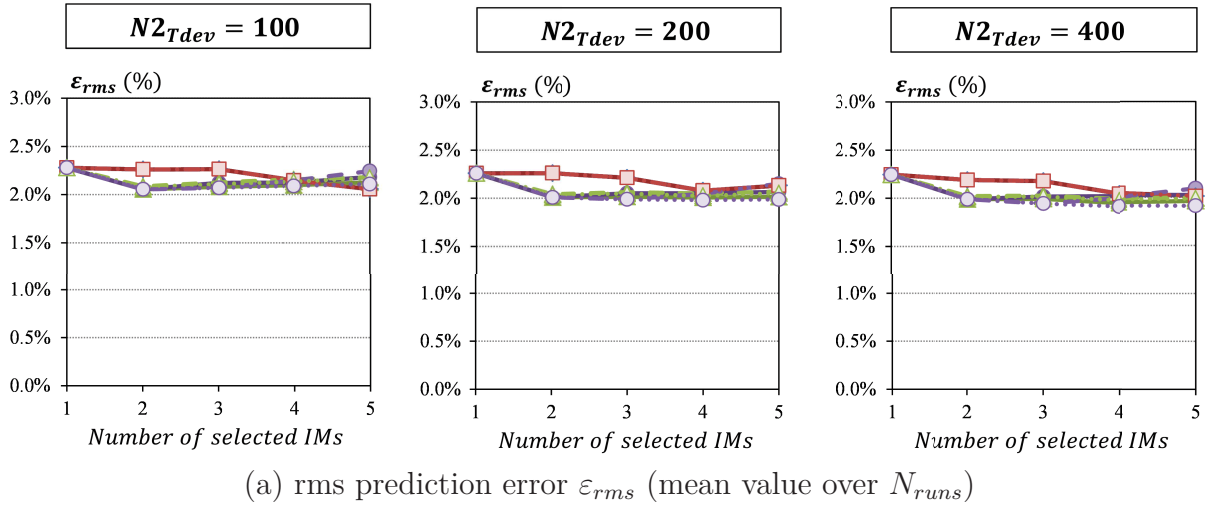


Figure 2.16: Prediction accuracy for the transceiver test vehicle considering different IM selection strategies and different sizes of training set

For the transceiver test vehicle, prediction accuracy achieved by the different models is less dependent on the different investigated factors than for the PA test vehicle. Indeed regard-

ing the rms prediction error, it varies only between 1.95% up to 2.25% and results are in good agreement with the expected accuracy, as estimated from model NRMSE computed on learning devices. Regarding maximal prediction error, dispersion is observed for models built on a training set of 100 or 200 devices, but significantly reduces when models are built on a training set of 400 devices. Similarly to the PA test vehicle, models built with 4 or 5 IMs can produce worse maximal error than models built with fewer IMs and best results are obtained when models are built on the largest training set. However contrarily to the PA test vehicle for which there is a huge difference between the rms and maximal prediction errors observed in the best case, the maximal prediction error observed here in the best case remains in a normal range, i.e. about 3 times the rms prediction error.

More generally this evaluation reveals one important weakness of the indirect test strategy, which offers good accuracy but limited reliability, i.e. even if most of the devices are predicted with low prediction error, some devices might suffer a rather large prediction error. Taking into account that the number of such devices is extremely small, the maximal prediction error is not a relevant metric to quantify prediction reliability. Instead, the Failing Prediction Rate $FPR(\epsilon_{meas})$ appears as a meaningful metric, since it quantifies the ratio of devices affected with a prediction error that exceeds the conventional measurement uncertainty. Figures 2.17 and 2.18 summarize prediction reliability results for the PA and transceiver test vehicles, respectively.

For the PA test vehicle, prediction reliability results clearly illustrate the inferior performances of the selection strategy based on Pearson correlation ranking compared to the three other selection strategies. The two wrapper methods lead to similar performances and the best performance is actually attained by using the selection strategy based on distance correlation ranking. In this case, a low FPR of about 0.35% is achieved for models built with 3 IMs, while wrapper methods lead to a FPR around 0.5%. Finally note that, although prediction accuracy is sensitive the size of the training set used during selection procedure and the size of the training set used to build the model, these parameters have a minor impact on prediction reliability for this test vehicle.

For the transceiver test vehicle, prediction reliability achieved by the different models is

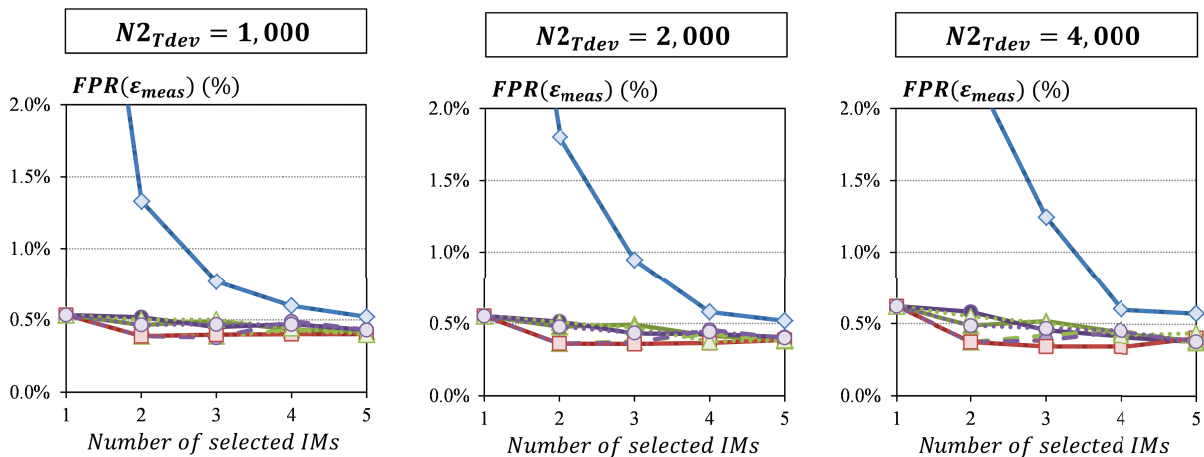


Figure 2.17: Prediction reliability for the PA test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})

actually influenced by many factors, i.e. the selection strategy, the number of selected IMs, the size of the training set used during selection procedure and the size of the training set used to build the model. Best performances are attained when performing IM selection with wrapper methods on a large training set, and building models with at least 3 IMs on a large training set. In this case, a low FPR of about 0.5% can be achieved.

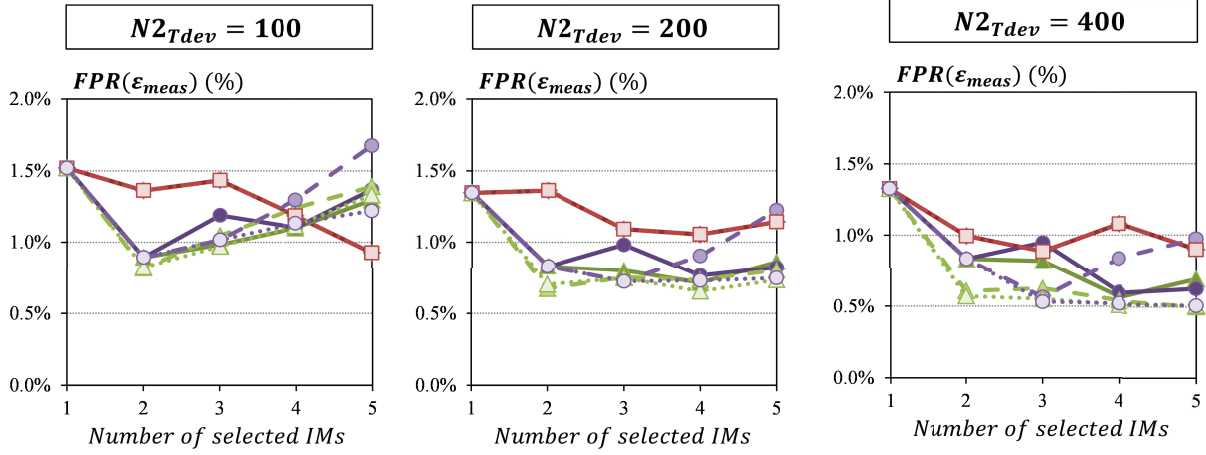


Figure 2.18: Prediction reliability for the transceiver test vehicle considering different IM selection strategies and different sizes of training set (mean value over N_{runs})

From this evaluation on two distinct test vehicles, we can derive some conclusions. First regarding the size of the training set, results show that best performances are indeed achieved when both IM selection and model construction are performed on a large training set. Then regarding the number of IMs used to predict the device performance, results show that although the intrinsic model accuracy is improved by adding more IMs to the model, this does not necessarily improve prediction reliability. The adequate choice of the number of IMs should therefore rely on the newly-introduced FPR metric rather than on the classical MSE metric. For the two case studies evaluated in this chapter, $N_{IM} = 3$ appears to be an appropriate choice. Finally regarding IM selection strategies, there is no evidence of one strategy being better than the others. Indeed, for the PA test vehicle the best performance is achieved by using a filter method, i.e. selection based on distance correlation ranking, while for the transceiver the best performance is achieved by using a wrapper method, i.e. selection using MARS built-in feature. These results are summarized in figure 2.19 which reports the FPR achieved for the two test vehicles using the different IM selection strategies, with $N1_{Tdev} = 80\% * N_{Tdev}$, $N2_{Tdev} = 80\% * N_{Tdev}$ and $N_{IM} = 3$. Note that in both cases, by choosing the appropriate IM selection strategy, good test efficiency can be achieved since more than 99.5% of the devices are accurately predicted with an error lower than the measurement repeatability error. Such high test efficiency permits to positively consider the implementation of the alternate testing, in particular for wafer-level testing where it offers the possibility to perform effective screening of defective devices at very low-cost.

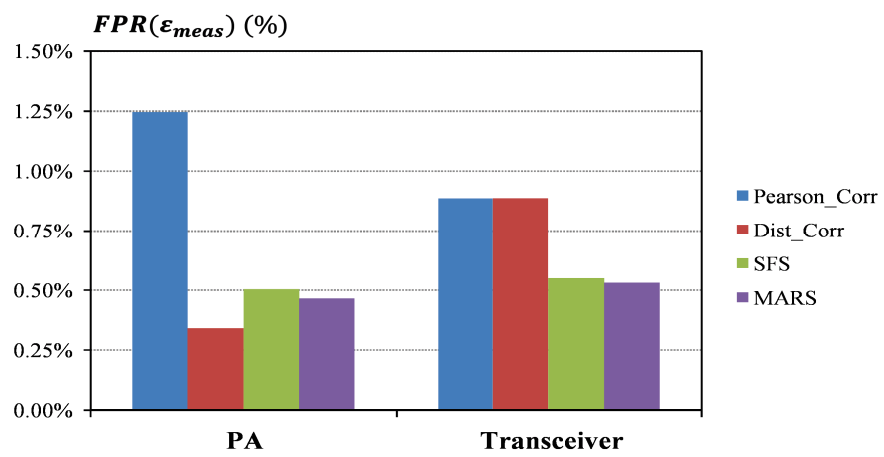


Figure 2.19: Comparative analysis of the different IM selection strategies for the two test vehicles

2.5 Summary

In this chapter we have focused on two aspects related to the classical implementation of the alternate test strategy, i.e. outlier filtering and IM selection. We have first presented an adaptive k-filter that permits to discard outlier circuits for the database. This filter iteratively prunes outliers by exploiting the sensitivity of descriptive statistics to outliers. It can handle IM space of high-dimensionality, it does not require information on defective devices and it does not rely on any assumption regarding the type of IM distribution. Then in the second part of the chapter, we have addressed the problem of IM selection, which is an essential step for the construction of efficient prediction models. We have explored different feature selection algorithms, with the objective to generate a single prediction model for each device performance. More specifically, we have performed a comparative analysis of four IM selection methods, pertaining to both filter and wrapper categories. We have also investigated the influence of several parameters such as the number of IMs used to perform prediction of a device performance, the size of the training set used during the IM selection procedure, and the size of the training set used for the model construction.

Efficiency has been evaluated in terms of model and prediction accuracy using classical metrics such as NRMSE, average and maximal prediction errors, but also in terms of prediction reliability using the FPR metric introduced in the first chapter. The study has been conducted using experimental data from two distinct RF test vehicles, i.e. a power amplifier and a transceiver.

Results have clearly pointed out that selection of indirect measurements is not an easy task that can be efficiently addressed by a single strategy. Indeed we have seen that selection based on distance correlation ranking leads to the best performance in case of the PA test vehicle, while it is the built-in feature of MARS algorithm in case of the transceiver test vehicle. These results reveal that finding the set of most appropriate IMs is strongly dependent on the case study and such analysis should be realized for every new case study. Results have also shown that with the appropriate selection strategy, good prediction reliability can be achieved using only a limited number of indirect measurements. In particular for both test vehicles, we found out that 99.5% of the devices can be accurately predicted with an error lower than the measurement repeatability error using only three indirect measurements.

Multi-model approach: Model generation

Contents

3.1 IM space reduction	54
3.1.1 PCA-based reduction	54
3.1.2 Pearson correlation-based reduction	57
3.1.3 Iterative MARS-based reduction	58
3.1.4 Preliminary evaluation of IM space reduction solutions	61
3.2 Multi-model generation	62
3.2.1 Extended SFS-Parental strategy	62
3.2.2 Extended SFS-Non Parental strategy	63
3.2.3 Computational effort	64
3.3 Evaluation	65
3.3.1 Model accuracy: evaluation on TS	65
3.3.2 Built models evaluation on VS	68
3.3.3 Further analysis and discussion	74
3.4 Summary	78

IM selection strategies presented in the previous chapter aim at generating a single prediction model for each device performance. In this chapter, we target the generation of several models for each device performance. The goal behind this is to implement a robust model redundancy strategy in view of reinforcing confidence in test predictions.

As for single model generation, the multi-model generation process relies on an exploration of the possible IM combinations from the available IM space. However, in order to perform efficient exploration, our idea is to implement a pre-processing step to preselect a limited number of pertinent indirect measurements. Indeed, it is very likely that the initial indirect measurement space contain non-relevant or noisy data that should not be used for the construction of a model. The objective is to keep in the search space only IMs that contain valuable information. By reducing the size of the search space in a preliminary phase, we can

then perform a more thorough exploration in a reasonable processing time.

This chapter is organized as follows. In section 3.1, we investigate different options for the reduction of the IM space. Then in section 3.2, we expose two different strategies for multi-model generation, called Parental and Non-Parental search strategies. At first, we will evoke the manner to construct the models by the mean of those two strategies. Finally in section 3.3, we perform evaluation of the generated models, in terms of both prediction accuracy and prediction reliability.

It has to be mentioned that for the experiments presented in this chapter on the two test vehicles, we will not investigate the influence of the size of the training set. Instead we choose a fixed training set size corresponding to 20% of the whole dataset and evaluation will be performed on the remaining 80%. This choice is motivated by the fact that the number of devices evaluated by the indirect test methodology during the production testing phase is evidently much higher than the number of devices used in the initial learning phase to establish the prediction models. Choosing a Validation Set (VS) larger than the Training Set (TS) therefore seems more representative of a practical situation.

3.1 IM space reduction

The objective of this section is to define an efficient procedure for the reduction of the IM space. This procedure will be used as a pre-processing step in the multi-model generation flow.

The motivation is to remove useless or misleading IMs in order to limit the size of the IM combination search space. Indeed, in case of high-dimension IM space, we can reasonably assume that the space includes IMs with minor relevance. By removing these IMs in a preliminary phase, we can significantly reduce the size of the IM combination search space. As an example, let us consider the transceiver test vehicle for which we have 405 IMs. Assuming that prediction models should not use more than 10 IMs to avoid curse of dimensionality issue, the size of the search space is $\sum_{i=1}^{10} C_i^{405} = 3 * 10^{19}$. If we reduce the IM space to only 30 relevant IMs, the size of the search space decreases down to $\sum_{i=1}^{10} C_i^{30} = 5.3 * 10^7$, so a reduction by a factor of more than 500 million. A more comprehensive exploration of this space can then be realized in a tractable processing time.

Several options are investigated in this section for the reduction of the IM space. Note that these options will be evaluated only on the transceiver test vehicle because it disposes of a high-dimension IM space (405 IMs). In case of the PA test vehicle, we have considered that the IM space (37 IMs) is small enough and then there is no need for the reduction process.

3.1.1 PCA-based reduction

The first option we have explored is based on Principal Component Analysis (PCA). PCA is a statistical procedure that can be used for dimensionality reduction [59]. The principle behind is a feature transformation. More precisely, it uses an orthogonal transformation to create a new set of linearly uncorrelated variables, called Principal Components (PC), from the initial set of possibly correlated variable. Each component is a linear combination of the original variables. This transformation is defined in such a way that the first principal component has

the largest variance and each succeeding component in turn has the highest variance under the constraint that it is orthogonal to the preceding components. This transformation maps a data vector from an original space of m variables to a new space of m variables which are uncorrelated over the dataset. Dimensionality reduction is obtained by keeping only the first L principal components.

As an illustration, figure 3.1 shows the Pareto scaling graph of the first ten principal components for a PCA transformation applied on the training set of the transceiver test vehicle. This figure shows that the first 9 components explain 92% of the total variance, which suggests that only these 9 PCs can be kept instead of the total 405 in the context of dimensionality reduction. However if this truncation indeed reduces the size of the new created space, it does not reduce the IM space since each PC involves all indirect measurements.

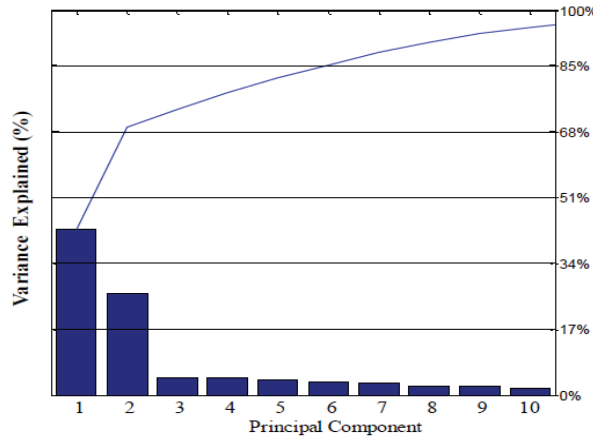


Figure 3.1: Pareto scaling graph for the first ten principal components

In this context, our proposal to perform IM space reduction based on PCA consists of:

- ◊ rank IMs in decreasing order of their weighting coefficient in each individual principal component,
- ◊ select only the highest-ranked IMs in the first PCs.

The first experiment we have performed involves the selection of IMs only in the first PC. Practically, we have built MARS models using an increasing number of IMs selected according to their weighting coefficient in the first PC and we have recorded the correlation coefficient computed on the training set for each one of these models. Figure 3.2 reports the evolution of this correlation coefficient as the number of IMs used in the model increases.

Note that we were not able to record till the 405 IMs because the simulation was time consuming; we have stopped it for 125 IMs (after 11 days).

As expected, the global tendency is that correlation improves by using more indirect measurements. However it also appears that models built with a limited number of IMs have a very poor correlation coefficient. For example if we consider the thirtieth model, i.e. the model built using the 30 highest-ranked IMs in the first PC, the correlation coefficient on

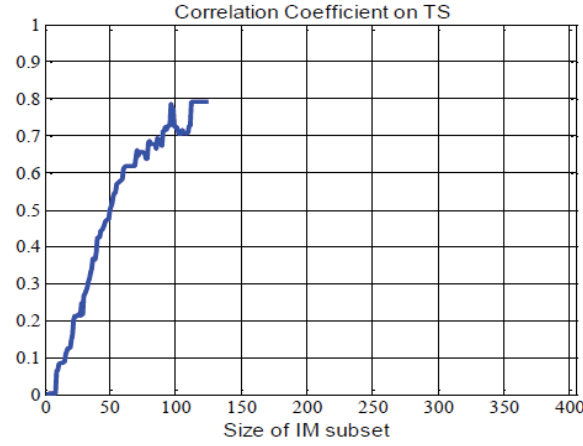


Figure 3.2: Correlation coefficient for MARS models built using an increasing number of IMs selected in the first PC

training set is under 0.3. As illustrated on the associated scatter graph given in figure 3.3, this model is not able to correctly present the correlation between the estimated and actual performances. Satisfying correlation coefficient above 0.7 is achieved only for models that use at least 90 IMs, which cannot be envisaged with respect to curse of dimensionality issue. Reduction of the IM space through selection in the first PC is therefore not a viable option.

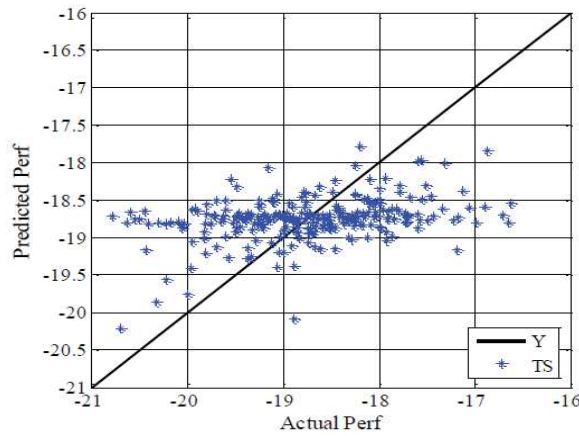


Figure 3.3: Scatter graph for model built using the 30 highest-ranked IMs in the first PC

Other experiments were performed considering more principal components, since the first PC only explains roughly two-thirds of the total variance. The objective is to select a subset of 30 IMs among the 405 available candidates. In particular, we have considered:

- selection in the first 3 PCs (which explain about 74% of the total variance): the 10 best-ranked IMs in PCA1, the 10 best-ranked IMs different from already selected ones in PCA2, and the 10 best-ranked IMs different from already selected ones in PCA3,

- selection in the first 9 PCs (which explain about 92% of the total variance): the 7 best-ranked IMs in PCA1, the 6 best-ranked IMs different from already selected ones in PCA1, the 5 best-ranked IMs different from already selected ones in PCA2 and so on until selecting only one IM in PCA7, PCA8 and PCA9.

To evaluate the viability of these options, MARS models have been built for these two selected IM subsets. Figure 3.4 reports the correlation coefficient computed on the training set for those models (model built using the 30 best-ranked IMs selected only in PCA1 has been included for the respect of comparison). These results show that even by increasing the number of principal components considered in the selection procedure, models build on the corresponding reduced IM space still exhibit poor correlation coefficient.

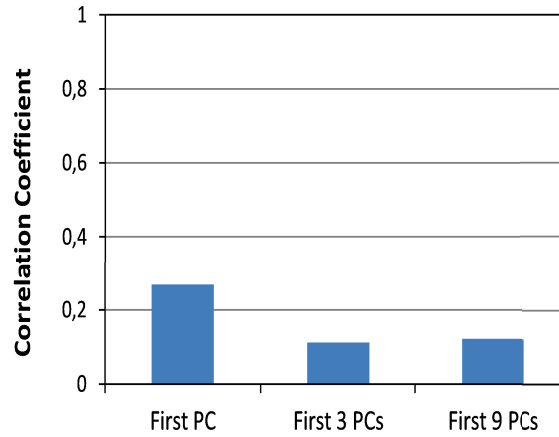


Figure 3.4: Correlation coefficient computed on training set for models built on reduced IM space using PCA-based selection

As a conclusion, PCA is a transformation technique that permits to obtain a new space with a reduced dimension able to represent the original space, but it does not permit to reduce the number of original variables. Options explored to perform selection of a reduced number of variables were not conclusive. Similarly, we have tried to perform IM space reduction based on Factor Analysis (FA), which is another transformation technique, but here again no conclusive results were obtained. Such feature transformation techniques therefore have no benefits in our context.

3.1.2 Pearson correlation-based reduction

The second option we have explored is based on the computation of sample Pearson correlation coefficient. The Pearson correlation coefficient is a statistical measure of the linear dependence between two variables [60]. More precisely, indirect measurements are sorted in decreasing order of their Pearson correlation coefficient to the targeted performance. IM space reduction is then simply performed by selecting the highest-ranked indirect measurements. This procedure has been applied to the transceiver test vehicle with the objective to select a

subset of 30 IMs among the 405 available candidates. A MARS model has been built using all the 30 IMs of this subset, which exhibit a very satisfying training correlation coefficient equal to 0.843. This result shows that Pearson correlation coefficient indeed permits to select a subset of relevant IMs compared to selection based on PCA. The main drawback of the approach is that IMs are considered independently, while combinations of IMs may provide significant information, which is not available at individual IM level.

3.1.3 Iterative MARS-based reduction

The third option we have explored is based on the built-in selection feature of the MARS algorithm. Indeed as introduced in the previous chapter, the number of IMs selected by the algorithm can actually be controlled by means of a user-configurable parameter, i.e. the `maxFuncs` parameter which specifies the maximal number of basis functions included in the model. Simulations were run trying to force the MARS algorithm to select 30 IMs among the 405 candidates. However the processing time is strongly dependent on the number of basis functions included in the model. For this case study, we still had no answer after several days of process.

To optimize the computational time, we have developed an indirect procedure based on an iterative use of the MARS built-in selection feature. Indeed since the limiting factor in terms of processing time is the number of IMs to be selected, the idea is to use the MARS algorithm in order to select only a reduced number q of IMs at each iteration; these selected IMs are then removed from the set of available IMs for the next iteration. The procedure stops when the size of the reduced IM space exceeds the target size of the reduced IM space. With q much smaller than the target size of the reduced IM space, selection is achieved in reasonable processing time. This procedure is illustrated in figure 3.5.

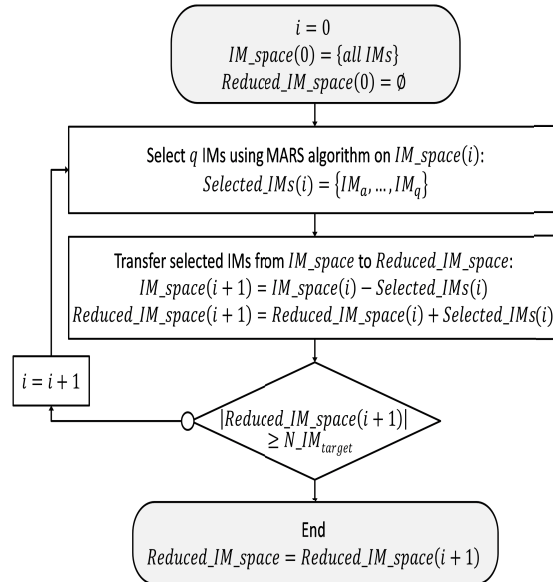
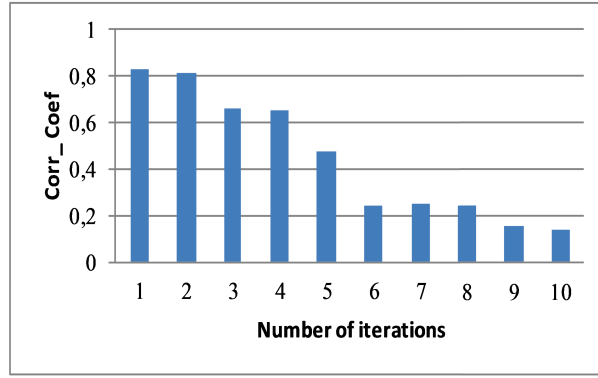


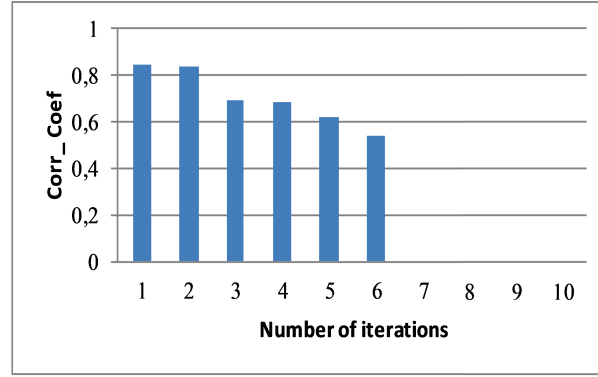
Figure 3.5: IM-space reduction based on MARS built-in selection feature

This procedure has been applied to the transceiver test vehicle with the objective to select a subset of 30 IMs among the 405 available candidates, considering three different values of the number of IMs selected at each iteration: $q = 3, 5$ and 10 . Note that the number of iterations performed depends on the number of IMs selected at each iteration. Ten iterations are necessary in case of 3-by-3 selection, six iterations in case of 5-by-5 selection and only three iterations in case of 10-by-10 selection. Figure 3.6 summarizes the correlation coefficient computed on the training set for the models built at the various iterations, for the three different values of q .

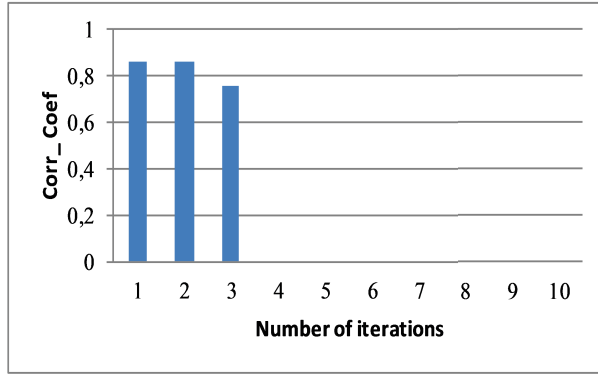
(a) 3-by-3 selection



(b) 5-by-5 selection



(c) 1-by-1 selection

**Figure 3.6:** Correlation coefficient for models built during iterative MARS-based selection

From figure 3.6, we can observe that whatever the value of q , models built in the first iterations present good correlation coefficient. As expected, the correlation coefficient reduces as the number of iteration increases. In particular, a very poor correlation coefficient is obtained for models built in the last iterations in case of 3-by-3 and 5-by-5 selection. However the correlation coefficient remains relatively high at the last iteration in case of 10-by-10 selection. Figure 3.7 reports the correlation coefficient computed on the training set for MARS models built using all the 30 IMs of the selected subsets. These results show that similar performance is achieved whatever the number q of IMs selected at each iteration, with a correlation coefficient in the same range than the one achieved using Pearson correlation-based selection.

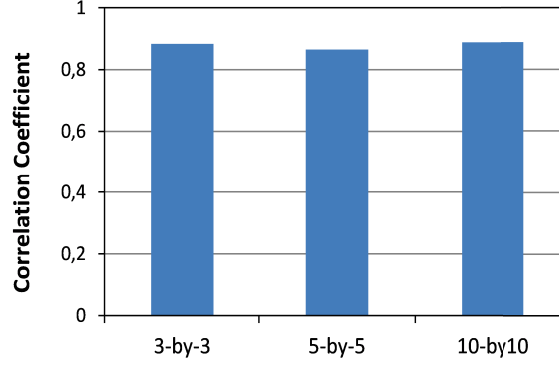


Figure 3.7: Correlation coefficient computed on training set for models built on reduced IM space using iterative MARS-based selection

3.1.4 Preliminary evaluation of IM space reduction solutions

To assess the validity of the different IM space reduction solutions, the MARS built-in selection feature has been used to select 5 IMs among the 30 available indirect measurements in the reduced IM spaces and the performances of the corresponding models have been evaluated. In addition, the algorithm has also been run the full original IM space on in order to obtain a reference model. Results are summarized in table 3.1, which gives the correlation coefficient computed on the training set together with the rms error.

As previously observed, these results show that preselection of pertinent IMs based on PCA

Table 3.1: Evaluation of IM space reduction solutions

		<i>Corr - Coef</i>	ε_{rms}
From PCA	RIM_{PCA1}	0.257	4.112
	$RIM_{PCA1-to-3}$	0.216	4.179
	$RIM_{PCA1-to-9}$	0.152	4.204
From Pearson Correlation	$RIM_{MARS-by-3}$	0.834	1.817
From MARS built-in selection feature	$RIM_{MARS-by-3}$	0.834	1.817
	$RIM_{MARS-by-5}$	0.841	1.780
	$RIM_{MARS-by-10}$	0.841	1.780
Full IM space (Reference)	FIM	0.841	1.779

is not a viable option since the correlation coefficient is very low. In contrast, all other options yield comparable results with a maximum degradation of the correlation coefficient compared to the model built on the full IM space of about 0.01% and similar rms error. We will therefore keep these options for the exploration of the IM combination space as detailed in the next paragraph and we will discard the IM spaces reduced by the mean of the PCA.

3.2 Multi-model generation

The multi-model generation process relies on the exploration of the IM combination space, taking into account a reduced IM space. As introduced in Chapter 2, the classical approach to perform this exploration in case of single-model generation is the SFS (Sequential Forward Selection) strategy, where IM combinations of increasing size are explored at each iteration, always keeping the best combination for the following iteration. The fundamental of our approach is based on a variation of the classical SFS strategy, in which several IM combinations are kept at each iteration. Two different options are exposed in this section for the choice of the IM combinations retained at each iteration; we have called those two approaches as extended SFS-Parental and extended SFS-Non_Parental strategies.

Note that although the primary objective of the extended SFS strategy is the generation of several accurate models for a given device performance, it can also be beneficial in the context of single model generation. Indeed since this strategy allows a larger exploration of the space than the classical SFS strategy, it may permit to identify a better IM combination.

3.2.1 Extended SFS-Parental strategy

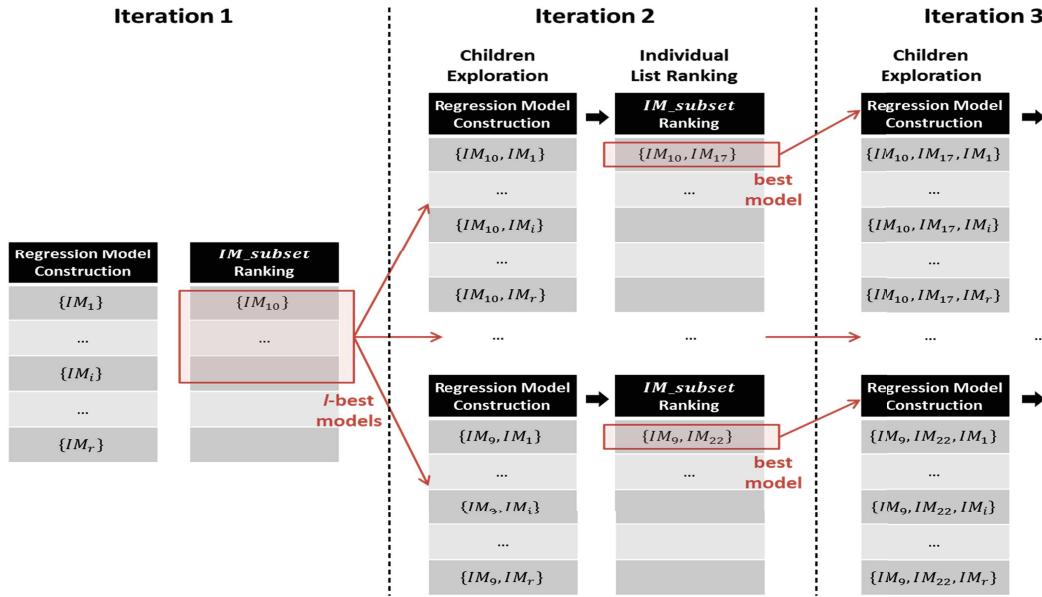


Figure 3.8: Extended SFS-Parental strategy

Figure 3.8 describes the extended SFS-parental strategy. Let us denote k the maximum number of IMs to be used by a prediction model to avoid curse of dimensionality issue and l the number of IM combinations retained at each iteration. The different steps performed are as follows:

1. A regression model is built using each individual IM of the reduced IM space and the rms error of each model is computed on the training set. IM are then sorted in increasing order of the rms error of their associated model and the l -best IMs are retained.

2. For each individual IM combination retained at the previous iteration, a regression model is built for every possible child by adding a new IM not already used in the IM combination and the rms error of each model is computed on the training set. l individual lists are therefore obtained. Children ranking is then performed in each one of the individual list according to the lowest model error and the best child of each list is retained.
3. The preceding step is repeated until the size of the retained IM combinations reaches the maximum number k of IMs to be used by a prediction model.

At the end of the procedure we have a total of $k * l$ lists of IM combinations, with l lists for every IM combination size from 1 to k . This strategy is called "parental" one because the link between parents and children is preserved between succeeding iterations. Indeed as illustrated in figure 3.9, all children present in the list of retained IM combinations at a given iteration have their parents present in the list of retained IM combinations at the preceding iteration. The interest of this parental method is that it pushes towards a diverse exploration of the space.

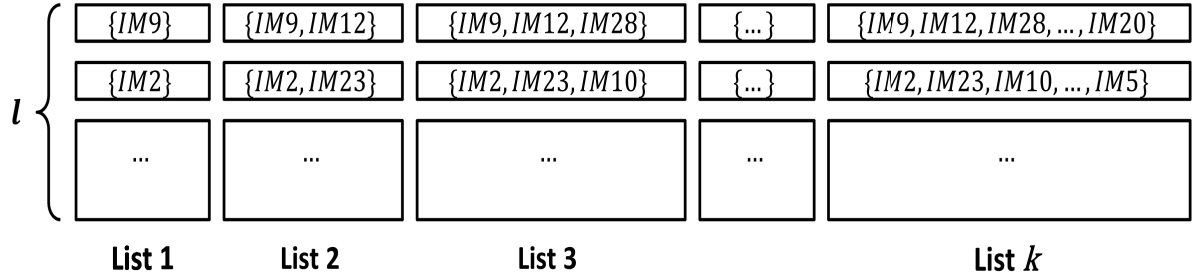


Figure 3.9: Extended SFS-Parental strategy: IM combination lists

3.2.2 Extended SFS-Non Parental strategy

The Non-Parental strategy has a similar process as the Parental strategy with some slight differences. We will no more pay attention to preserving the link between parents and children between succeeding iterations. For this reason we have called the strategy "Non-Parental".

Figure 3.10 describes the different steps of the Non-Parental strategy. They are as follows:

1. A regression model is built using each individual IM of the reduced IM space and the rms error of each model is computed on the training set. IM are then sorted in increasing order of the rms error of their associated model and the l -best IMs are retained.
2. For each individual IM combination retained at the previous iteration, a regression model is built for every possible child by adding a new IM not already used in the IM combination and the rms error of each model is computed on the training set. The l individual lists are merged in a single list. Children ranking is then performed in this global list according to the lowest model error and the l -best children are retained.

3. The preceding step is repeated until the size of the retained IM combinations reaches the maximum number k of IMs to be used by a prediction model.

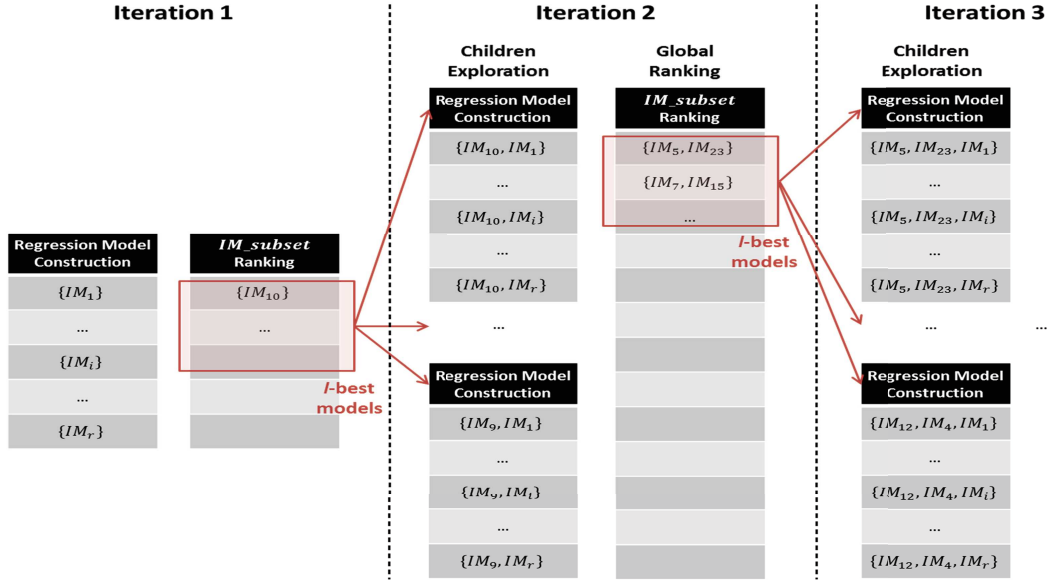


Figure 3.10: Extended SFS-non Parental strategy

Here again this procedure gives a total of $k * l$ lists of IM combinations, with l lists for every IM combination size from 1 to k . The main difference with the previous strategy is that the link between parents and children is not necessarily preserved between succeeding iterations. Indeed as illustrated in figure 3.11, some IM combinations present in the list at a given iteration may not have any child in the list of retained combinations at the following iteration while others may have several children. We therefore expect that this strategy leads to less diversity than the Parental strategy.

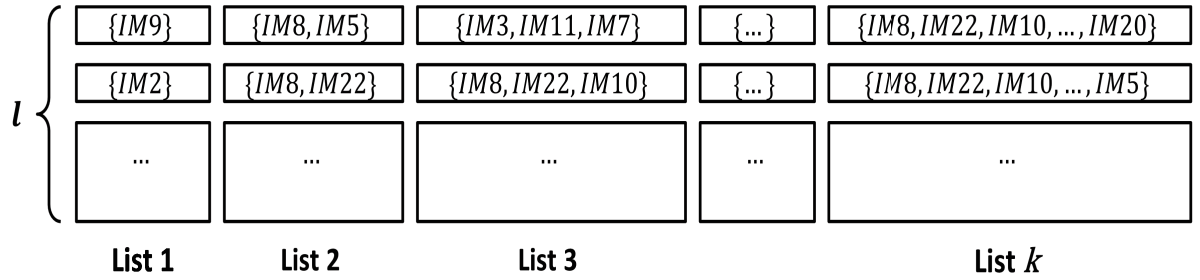


Figure 3.11: Extended SFS-Non Parental strategy: IM combination lists

3.2.3 Computational effort

The computational cost of the extended SFS strategy depends on the size r of the search space (reduced IM space), on the number l of retained subsets at each iteration, and on the

maximum size k of the retained subsets. The number of models to be built with this strategy is given by:

$$M_{extended-SFS} = l * \left(\sum_{i=1}^k C_{r-i+1}^1 - r \right) + r = l * \left(k * (r + 1) - \frac{k * (k + 1)}{2} - r \right) + r \quad (3.1)$$

As an example, let us consider the transceiver test vehicle for which we have a reduced IM space composed of $r = 30$ IMs (selected among the 405 IMs of the original space). Assuming that prediction models should not use more than $k = 10$ IMs to avoid curse of dimensionality issue and considering $l = 10$ subsets retained at each iteration, the extended SFS strategy lead to the construction of $M_{extended-SFS} = 2280$ models, which is easily tractable in a small processing time.

Note that the application of the extended SFS strategy on the original IM space of 405 IMs would require the construction of more than 36 thousands models. The IM space reduction performed as a pre-processing step therefore permits to gain a 15x speed-up in the processing time.

3.3 Evaluation

In this section, we evaluate the proposed strategy for multi-model generation on the two test vehicles. In case of the transceiver test vehicle, we have a large initial IM space composed of 405 candidates. We will therefore use the reduced IM spaces composed of 30 IMs obtained either from Pearson correlation ranking (denoted $RIM_{Pearson}$) or iterative use of MARS built-in selection feature (denoted $RIM_{MARS-by-q}$) with q the number of IMs selected at each iteration) for the exploration of the IM combination space. In case of the PA test vehicle, we have an initial IM space composed of only 37 candidates. We will therefore omit IM space reduction and directly perform the exploration of the IM combination space considering all available 37 candidates.

Results are presented first in terms of model accuracy evaluated on the training set, and then in terms of prediction accuracy and reliability evaluated on the validation set. Considered metrics are the rms error ε_{rms} , the maximal error ε_{max} and the failing prediction rate FPR, as defined by equations (1.4) to (1.6) in chapter I. These results are then discussed in the last part, analyzing the influence of the different solutions for IM space reduction and the two various options for multi-model generation.

3.3.1 Model accuracy: evaluation on TS

The proposed methodology for multi-model generation has been applied considering $l = 10$ IM combinations retained at each iteration of the procedure. Practically, both versions of the extended SFS strategy have been implemented, considering IM combinations composed of up to $k = 10$ IMs. For each selected combination, model accuracy has been evaluated by computing the rms error ε_{rms} on the training set. For the sake of comparison, we have also generated regression models using the built-in selection feature of the MARS algorithm run on the full IM space. These models are denoted "reference" models since they correspond to

models obtained by the classical approach.

Results are illustrated in figures 3.12 and 3.13, which give the rms training error according to the size of the selected IM combinations, for the transceiver and PA test vehicles respectively. Dots correspond to models built using the 10 retained IM combinations, for each combination size. For the transceiver test vehicle, curves link the best generated models for each combination size, considering a given reduced search space. These results are summarized in tables 3.2 and 3.3, which give the mean, minimum and maximum values of the rms training error of the generated models in comparison with the rms training error of the reference model.

Different remarks arise from those results: First as expected, one can observe the general trend of accuracy improvement when increasing the number of IMs used to build the models, with an rms error that reduces from about 1.95% down to about 1.65% by increasing the number of IMs from 2 up to 10 for the transceiver test vehicle, and from about 0.90% down to about 0.55% for the PA test vehicle. Still, the interesting point is that the proposed methodology actually permits to generate several models with a similar accuracy than the reference models generated on the full IM space, and even better in some cases. As shown in table 2 for the transceiver test vehicle, the difference between the mean value of the rms training error for the generated models and the rms training error for the reference model is about 0.025% and does not exceed 0.068%. Similarly as shown in table 3 for the PA test vehicle, this difference is about 0.045% and does not exceed 0.124%.

Regarding the different solutions investigated for IM space reduction in case of the transceiver test vehicle, the solution based on Pearson correlation appears less efficient than solutions based on the use of MARS built-in selection feature. Best results are actually obtained when the search is performed on the reduced IM space obtained from iterative use of MARS built-in selection feature with 10 IMs selected at each iteration ($RIM_{MARS-by-10}$). In this case, the best generated model has a lower rms training error than the reference model, whatever the IM combination size.

Table 3.2: Rms training error of generated models for the transceiver test vehicle

		ε_{rms} (%) on training set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	2,014	1,900	1,837	1,795	1,754	1,750	1,729	1,694	1,679
	min	1,942	1,858	1,804	1,769	1,710	1,714	1,679	1,664	1,634
	max	2,195	1,948	1,871	1,832	1,804	1,794	1,794	1,766	1,741
Reference model		1,946	1,860	1,819	1,779	1,754	1,733	1,709	1,678	1,652

Finally regarding the two different strategies for multi-model generation, it can be remarked that the dispersion observed between the 10 models is lower when using the non-parental method than the parental one. However, the diversity of IMs used in these models is lesser; for instance for the transceiver test vehicle, considering IM combinations composed of 5 IMs selected in $RIM_{MARS-by-10}$, the 10 models generated by the non-parental method involve only 9 IMs out of the 30 candidates while the 10 models generated by the parental method involve 14 IMs out of the 30 candidates.

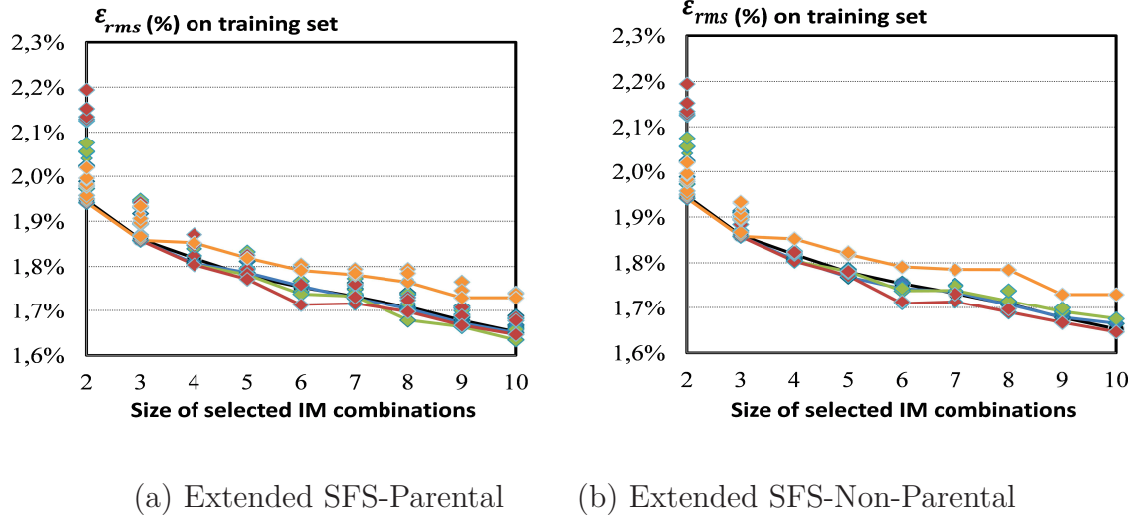


Figure 3.12: Model accuracy for the transceiver test vehicle

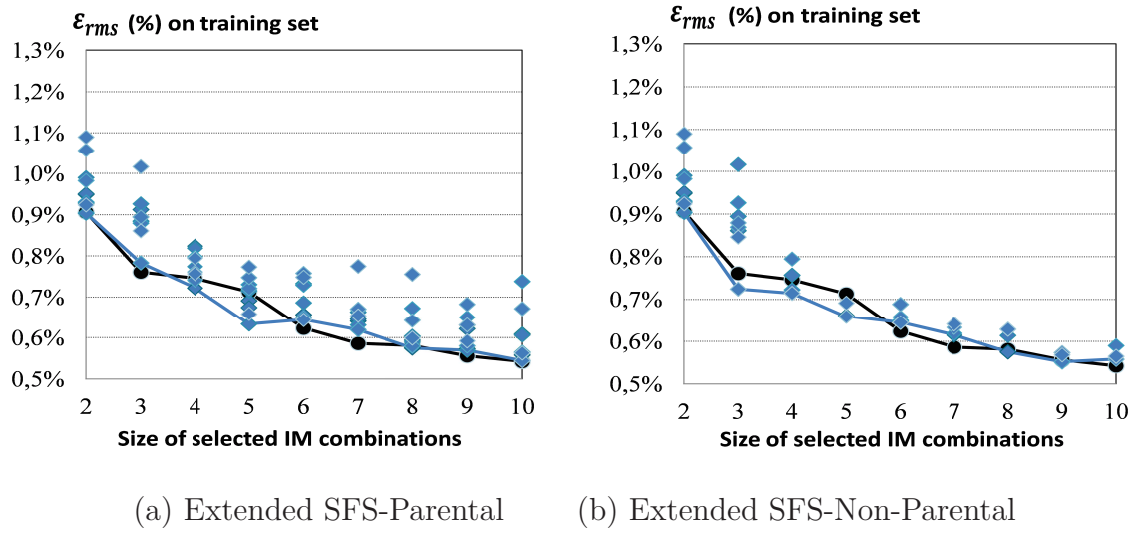


Figure 3.13: Model accuracy for the PA test vehicle

Table 3.3: Rms training error of generated models for the PA test vehicle

		ε_{rms} (%) on training set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	0,971	0,885	0,761	0,680	0,677	0,648	0,622	0,594	0,583
	min	0,904	0,724	0,714	0,633	0,644	0,614	0,575	0,552	0,545
	max	1,090	1,016	0,824	0,773	0,758	0,775	0,756	0,683	0,739
Reference model		0,906	0,761	0,746	0,713	0,623	0,586	0,581	0,556	0,542

3.3.2 Built models evaluation on VS

In order to evaluate prediction accuracy and reliability, generated models have been used to perform performance prediction for all devices of the validation set. Prediction accuracy is evaluated both in terms of rms and maximal errors, and prediction reliability is evaluated in terms of failing prediction rate.

Prediction accuracy results for both test vehicles

Figures 3.14 and 3.15 illustrate prediction accuracy results for the transceiver and PA test vehicles respectively. Tables 3.4 to 3.7 summarize the mean, minimum and maximum values of prediction errors in comparison with reference model, for the different sizes of selected IM subsets.

Analyzing these results, a first comment arises: contrarily to the monotonic trend observed when evaluating accuracy on the training set, the rms error does not necessarily reduces when increasing the number of IMs used to build the models. This is also expected from the fact related to the curse of dimensionality. For the transceiver and PA test vehicle, the lowest rms error is actually obtained for models built respectively with 3 and 4 IMs. Referring to table 4 for the transceiver test vehicle, the difference between the mean value of the rms error and the reference one remains below 0.1% whatever the size of selected IM subsets. Referring to table 6 for the PA test vehicle, this difference remains below 0.3% for subsets up to 8 IMs, and then slightly increases but does not exceed 1%. Those results demonstrate that the generated models are in the same range of prediction accuracy as the reference model. Also the computed minimum rms errors show that we are able to build models with prediction accuracy better than the reference one for both test vehicles, especially when increasing the number of selected IMs: an rms error lower by 0.1% can be obtained for the transceiver in case of a model built with 10 IMs, and an rms error lower by 0.2% for the PA in case of a model built with 8 IMs.

Regarding the maximal prediction error, no significant trend can be drawn. For the transceiver test vehicle, most of the generated models exhibit a maximal prediction error around 8% whatever the size of selected IM subsets, in the same range than the reference model. For the PA test vehicle, a large dispersion is observed: the maximal prediction error that can be lower than the reference one by more than 3% for some models, but it can also be higher by

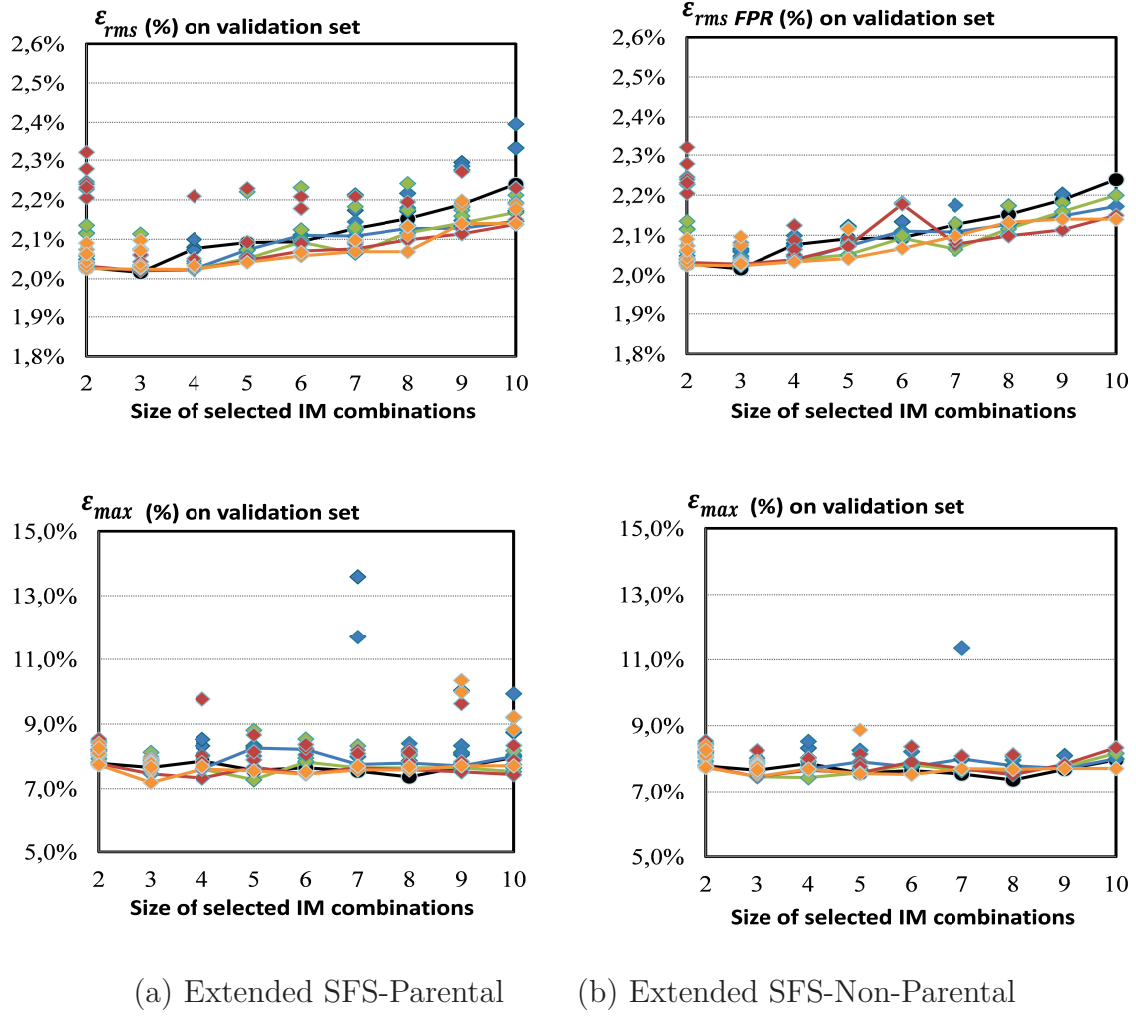
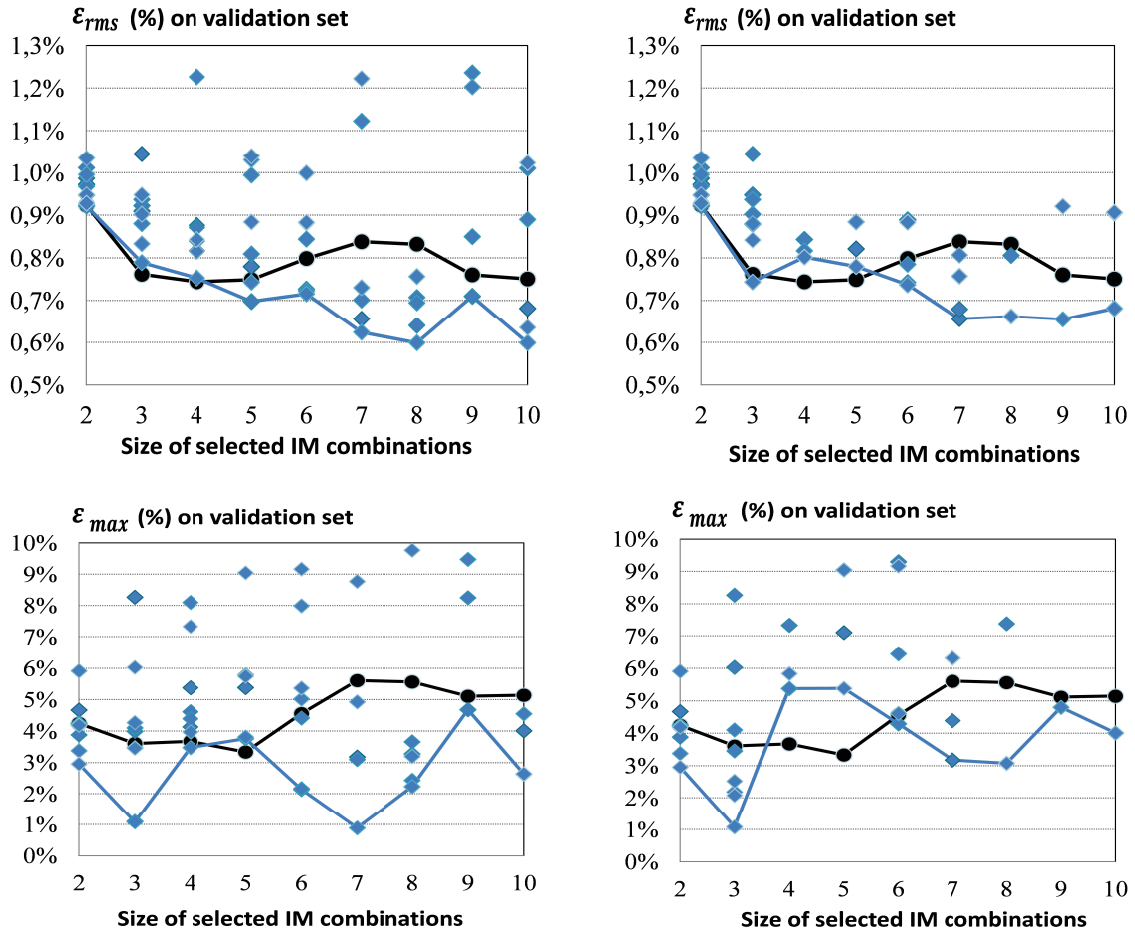


Figure 3.14: Prediction accuracy for the transceiver test vehicle

Table 3.4: Rms prediction error of generated models for the transceiver test vehicle

		ϵ_{rms} (%) on validation set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	2,090	2,047	2,053	2,068	2,113	2,117	2,136	2,160	2,182
	min	2,026	2,021	2,023	2,042	2,058	2,065	2,069	2,114	2,138
	max	2,320	2,114	2,209	2,229	2,231	2,213	2,242	2,295	2,396
Reference model		2,029	2,016	2,077	2,092	2,094	2,127	2,152	2,188	2,240



(a) Extended SFS-Parental (b) Extended SFS-Non-Parental

Figure 3.15: Prediction accuracy for the PA test vehicle

Table 3.5: Maximal prediction error of generated models for the transceiver test vehicle

		ϵ_{max} (%) on validation set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	8,063	7,771	7,823	7,913	7,967	8,361	7,896	7,930	8,132
	min	7,740	7,193	7,330	7,279	7,451	7,581	7,521	7,523	7,437
	max	8,537	8,253	9,774	8,874	8,536	13,597	8,401	10,352	9,938
Reference model		7,786	7,654	7,858	7,568	7,644	7,543	7,364	7,681	7,963

Table 3.6: Rms prediction error of generated models for the PA test vehicle

		ε_{rms} (%) on validation set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	0,972	0,913	0,994	0,900	0,949	1,119	0,955	1,702	1,389
	min	0,922	0,743	0,753	0,697	0,715	0,624	0,599	0,655	0,599
	max	1,035	1,045	2,279	1,642	1,819	2,143	2,118	3,822	4,564
Reference model		0,924	0,761	0,744	0,749	0,798	0,838	0,832	0,760	0,750

Table 3.7: Maximal prediction error of generated models for the PA test vehicle

		ε_{max} (%) on validation set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	4,195	4,191	6,991	7,952	8,867	10,508	8,493	22,344	15,735
	min	2,954	1,087	3,477	3,763	2,145	0,891	2,234	4,682	2,633
	max	5,917	8,266	18,100	22,617	25,685	30,994	30,994	57,313	69,006
Reference model		4,246	3,607	3,673	3,332	4,561	5,607	5,564	5,113	5,145

more than 50% for some other models (especially for models built with a high number of IMs).

However here again, interesting points using the proposed methodology are: (i) we are able to identify models with a better accuracy than the reference model generated on the full IM space and (ii) we have at our disposal not only a single model but several models with an accuracy at least equal to the reference one.

Prediction reliability results for both test vehicles

Figures 3.16 and 3.17 illustrate prediction reliability results for the transceiver and PA test vehicles respectively. Tables 3.8 and 3.9 summarize the mean, minimum and maximum values of failing prediction rates in comparison with reference model, for the different sizes of selected IM subsets.

A first general comment is that, as for prediction accuracy, prediction reliability does not necessarily improve when increasing the number of IMs used to build the models. This is clearly illustrated for the transceiver test vehicle, for which the FPR achieved by the reference model built with 10 IMs is more than twice higher than the FPR achieved by the reference model built with only 3 IMs.

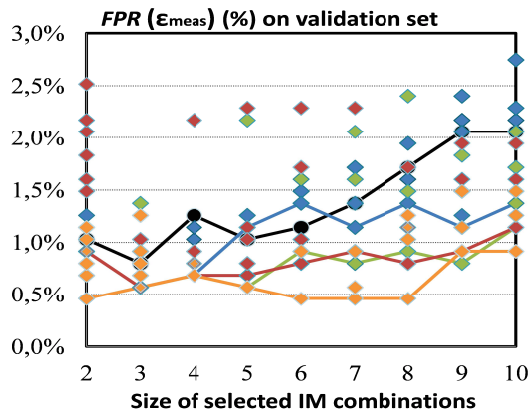
Then focusing more specifically on the comparison between the conventional approach and the proposed methodology, these results clearly reveal the benefit of the proposed methodology which permits to generate models with improved prediction reliability. Indeed referring to table 8, the best FPR achieved by a reference model for the transceiver test vehicle is around 0.8%; it can be reduced down to 0.46% using the proposed methodology, which corresponds to an improvement by about a factor 2. In the same way referring to table 9, the best FPR achieved by a reference model for the PA test vehicle is around 0.21%; it can be reduced down to 0.09%, which corresponds to an improvement by a factor of more than 2. Moreover, it is worth noting that most of the generated models have an FPR that outperforms the one of the reference model for a given size of selected IM subset, as illustrated in figures 8 and 9. Those facts reveal the ability of the proposed methodology to generate several reliable models.

Table 3.8: Failing prediction rate of generated models for the transceiver test vehicle

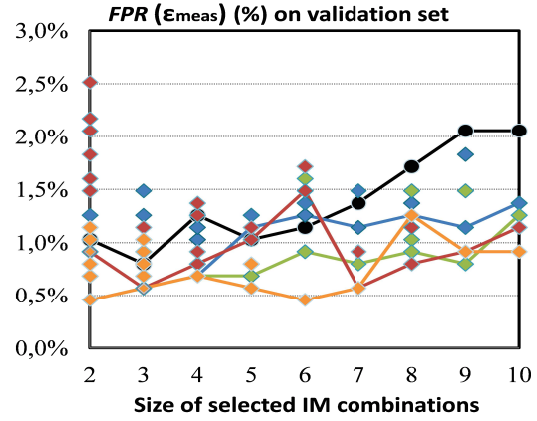
		$FPR(\varepsilon_{meas})$ (%) on validation set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	0,469	0,400	0,290	0,247	0,254	0,234	0,233	0,209	0,180
	min	0,335	0,257	0,212	0,179	0,212	0,145	0,157	0,089	0,123
	max	0,604	0,581	0,391	0,403	0,313	0,391	0,425	0,358	0,380
Reference model		0,425	0,526	0,503	0,335	0,246	0,313	0,335	0,268	0,212

Table 3.9: Failing prediction rate of generated models for the PA test vehicle

		$FPR(\varepsilon_{meas})$ (%) on validation set								
Size of selected IM comb		2	3	4	5	6	7	8	9	10
Generated models	mean	0,469	0,400	0,290	0,247	0,254	0,234	0,233	0,209	0,180
	min	0,458	0,572	0,686	0,572	0,458	0,458	0,458	0,801	0,915
	max	2,517	1,487	2,174	2,288	2,288	2,288	2,403	2,403	2,746
Reference model		1,030	0,801	1,259	1,030	1,144	1,373	1,716	2,059	2,059



(a) Extended SFS-Parental



(b) Extended SFS-Non-Parental

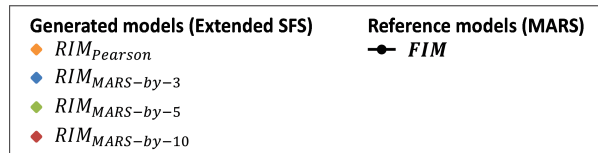
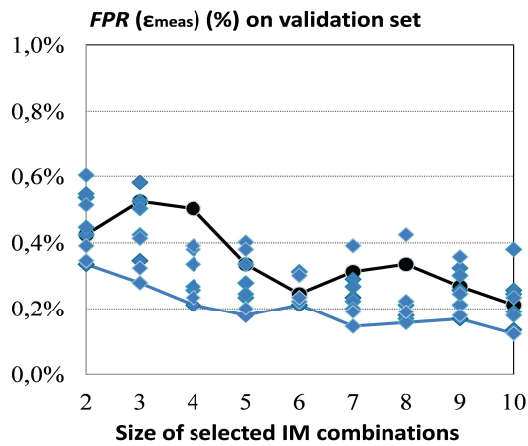
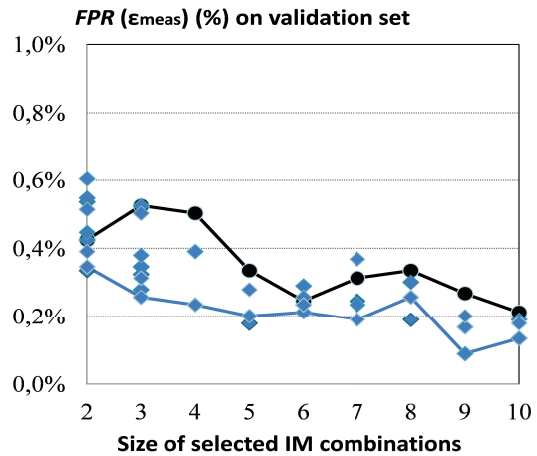


Figure 3.16: Prediction reliability for the transceiver test vehicle



(a) Extended SFS-Parental



(b) Extended SFS-Non-Parental



Figure 3.17: Prediction reliability for the PA test vehicle

3.3.3 Further analysis and discussion

In order to perform a comparative study of the different options for IM space reduction and multi-model generation, a detailed analysis has been conducted based on failing prediction rate results.

Performances of IM space reduction options

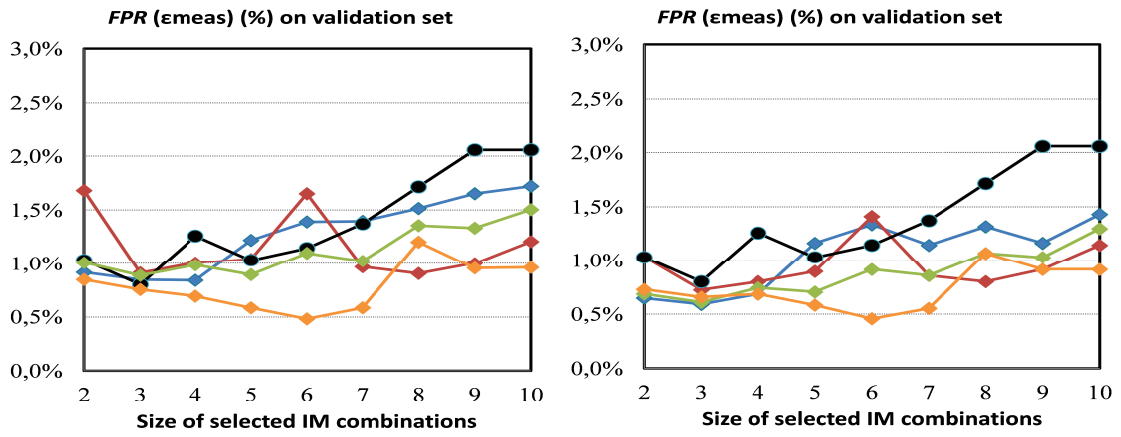
First we have investigated the influence of the option used for IM space reduction. This analysis is conducted only for the transceiver test vehicle since the initial IM space of the PA test vehicle is small enough to directly apply the multi-model generation procedure on the full IM space.

For the transceiver test vehicle, we have four different reduced IM spaces composed of 30 IMs obtained from either Pearson correlation ranking ($RIM_{Pearson}$) or iterative use of MARS built-in selection feature ($RIM_{MARS-by-3}$, $RIM_{MARS-by-5}$, $RIM_{MARS-by-10}$). For each one of these reduced spaces, we have a total of 180 generated models corresponding to 20 different models generated by parental and non-parental strategies for each IM combination size from 2 to 10.

Figures 3.18.a and 3.18.b report the mean FPR value over the 20 different models and the mean of the 3 best models according to the IM combination size, for the different IM space reduction options. The FPR value of models obtained using the built-in selection feature of the MARS algorithm applied on the full IM space is also reported as reference.

From both figures, we can note:

- Whatever the option used for IM space reduction, we have an improved reliability with respect to reference models generated on the full IM space. This confirms the validity of investigated options that indeed permit to preselect a limited number of pertinent IMs and allow the generation of efficient models, not only in terms of prediction accuracy but also in terms of prediction reliability.
- The best option for this test vehicle is IM space reduction based on Pearson correlation ranking. Note that this result should not be considered as a generic result since it just concerns one case study. However it is interesting to note that for this case study, the use of Pearson correlation ranking was not the more efficient method for IM selection in the context of single model generation (cf. Chapter 2); however it appears very efficient for preselection of pertinent IMs in the context of multiple model generation.



(a) mean over all generated models (b) mean over the best 3 models

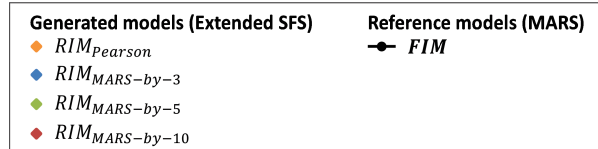


Figure 3.18: Influence of IM space reduction options on prediction reliability for the transceiver test vehicle

Performances of extended-SFS options

Then we have investigated the influence of the option used in the extended-SFS algorithm, i.e. parental and non-parental strategies. Results are summarized in figures 3.19 and 3.20 for the transceiver and PA test vehicles respectively. These figures report the mean FPR value over the different generated models and the mean of the 3 best models according to the IM combination size, for the two different strategies.

From these figures, we can note:

- There is no significant difference between parental and non-parental methods in terms of prediction reliability results since similar FPR performances are achieved for a given size of selected IM subsets, for both test vehicles.
- In case of the transceiver test vehicle, there is a substantial reduction of the mean FPR value when computed over the best 3 models, compared to the mean FPR value computed over all the generated models. This reveals that for a give size of selected IM subsets, all generated models are not equivalent in terms of prediction reliability, i.e. some have better performance than the over. It also reveals that it exists at least 3 models that yield to significant FPR improvement compared to reference models.
- In case of the PA test vehicle, the same trend can be observed but in a lessened extent. In this case, there is only a slight reduction between the mean FPR value computed over the 3 best models and the mean FPR computed over all the generated models. Still, there is a clear improvement compared to reference models, especially in case few IMs are used to build the models.

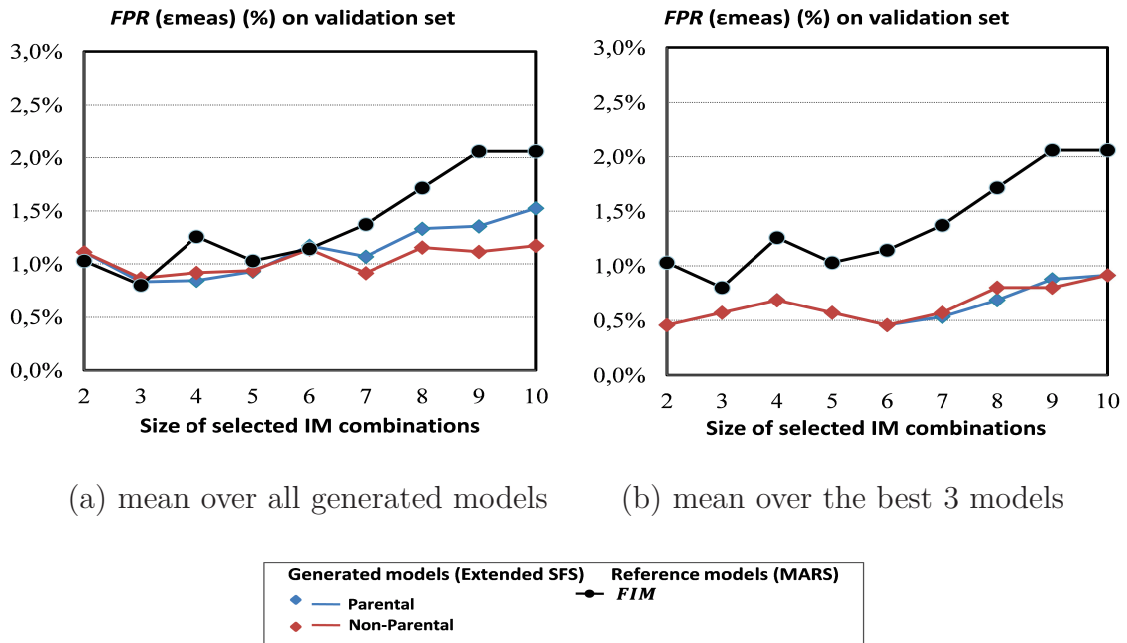


Figure 3.19: Influence of extended-SFS options on prediction reliability for the transceiver test vehicle

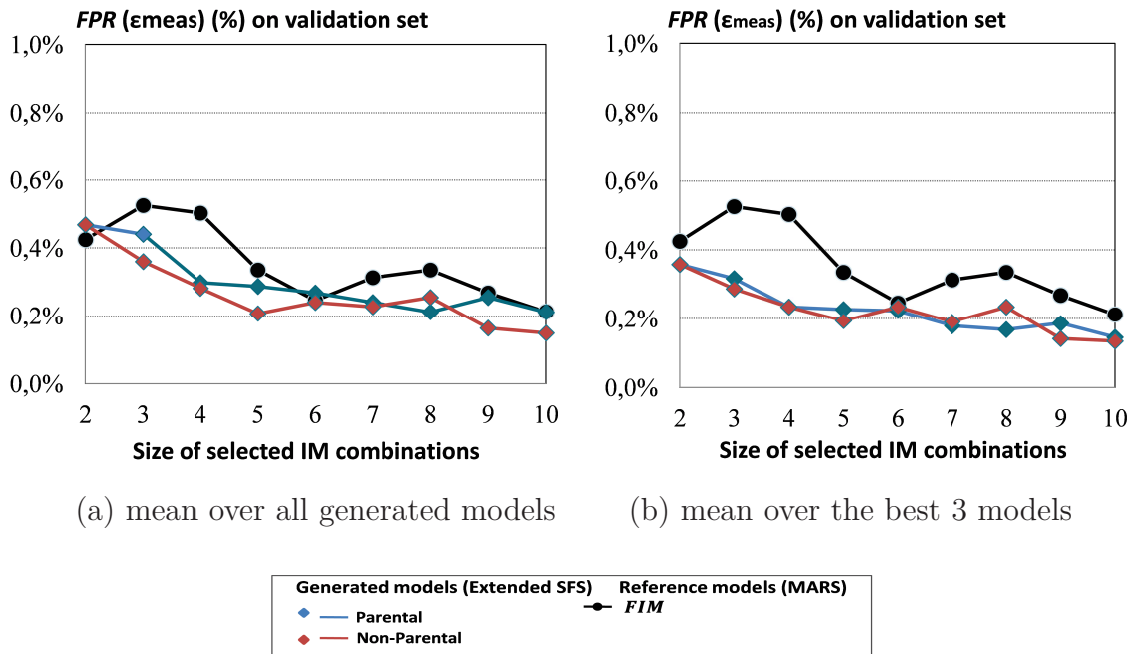


Figure 3.20: Influence of extended-SFS options on prediction reliability for the PA test vehicle

3.4 Summary

In this chapter, we have presented an original methodology for building efficient prediction models from a large set of indirect measurement candidates. Compared to state-of-the-art, the key idea is to perform a more thorough exploration of the IM combination space with the objective to (i) allow the generation of several accurate prediction models and (ii) possibly identify a better model than with the conventional method.

In order to permit a thorough exploration of the IM combination space in a tractable computing time, the methodology involves as first pre-processing step in which a reduced number of promising candidates are preselected from the full IM space. Thanks to this IM space reduction, we can afford to perform an extended search and we have developed two extended versions of the classical SFS algorithm, using either a parental or non-parental strategy. Both versions allow the selection of multiple IM subsets, and therefore the generation of several prediction models.

The methodology has been applied on the two test vehicles for which we have experimental data (the pre-processing step of IM space reduction have been omitted for the PA test vehicle because the initial IM space is small enough). Obtained results demonstrate that the proposed methodology permits to generate models with enhanced performance compared to the conventional solution. More specifically, we have seen that it permits to improve prediction reliability results by a factor of about 2. It also permits to generate not only a single efficient model, but several accurate and reliable models. The availability of several efficient models for each device performance actually opens the way for further alternate test developments, such as optimization over the different performances in order to reduce the global number of required indirect measurements, or the implementation of alternate test with model redundancy in order to improve test confidence. This latter aspect is developed in the next chapter.

Multi-model approach: Models redundancy

Contents

4.1	Model redundancy principle	80
4.2	Generic framework	83
4.2.1	Overview	83
4.2.2	Selection and construction of redundant models	84
4.2.3	Tradeoff exploration: reliability vs. cost	86
4.3	Results	87
4.3.1	Selection and construction of redundant models	87
4.3.2	Tradeoff between test cost and test reliability	90
4.4	Summary	94

AN essential point for the utilization of alternate testing deals with confidence in test predictions. Developments presented so far in this manuscript have permitted to propose a relevant metric allowing to quantify prediction reliability and an original methodology for the generation of efficient prediction models. Based on this methodology, we develop in this chapter a generic framework for efficient implementation of alternate test with model redundancy. The objective is to reinforce confidence in test prediction by exploiting model redundancy. Different options for the selection and construction of redundant models are implemented. As a consequence, the proposed framework will contain a set of solutions leading to different trade-offs between test cost and test reliability. This framework therefore helps the test engineer to choose the more appropriate solution with respect to his specific application context.

The chapter is organized as follows. Section 4.1 briefly reviews previous works related to confidence improvement of alternate testing and introduces the principle of model redundancy. Section 4.2 details the proposed framework for the implementation of alternate test with model redundancy. This framework is then used on the PA test vehicle in section 4.3 which

provides illustrative results.

Note that the proposed framework is evaluated only for the PA test vehicle as we have noticed a difference in the model performances when computed on the Training Set (TS) and on the Validation Set (VS), contrarily to the transceiver test vehicle. This is illustrated in figure 4.1, which reports the FPR metric computed either on TS and VS for the best models generated by the extended-SFS procedure, for the two test vehicles. In case of the PA test vehicle, prediction reliability results evaluated on the validation set are not as good as the ones evaluated on the training set. Indeed whatever the size of the selected IM combinations, FPR computed on VS is higher than FPR computed on TS. This suggests that prediction reliability improvement is possible for this test vehicle and the idea is to investigate whether the implementation of model redundancy can enhance FPR computed on VS close FPR computed on TS. In contrast for the transceiver test vehicle, FPR values computed on VS are very close to the ones computed on TS whatever the size of selected IM combinations, and even better in some cases. From this fact, we think that there is no improvement behind the implementation of model redundancy on this test vehicle.

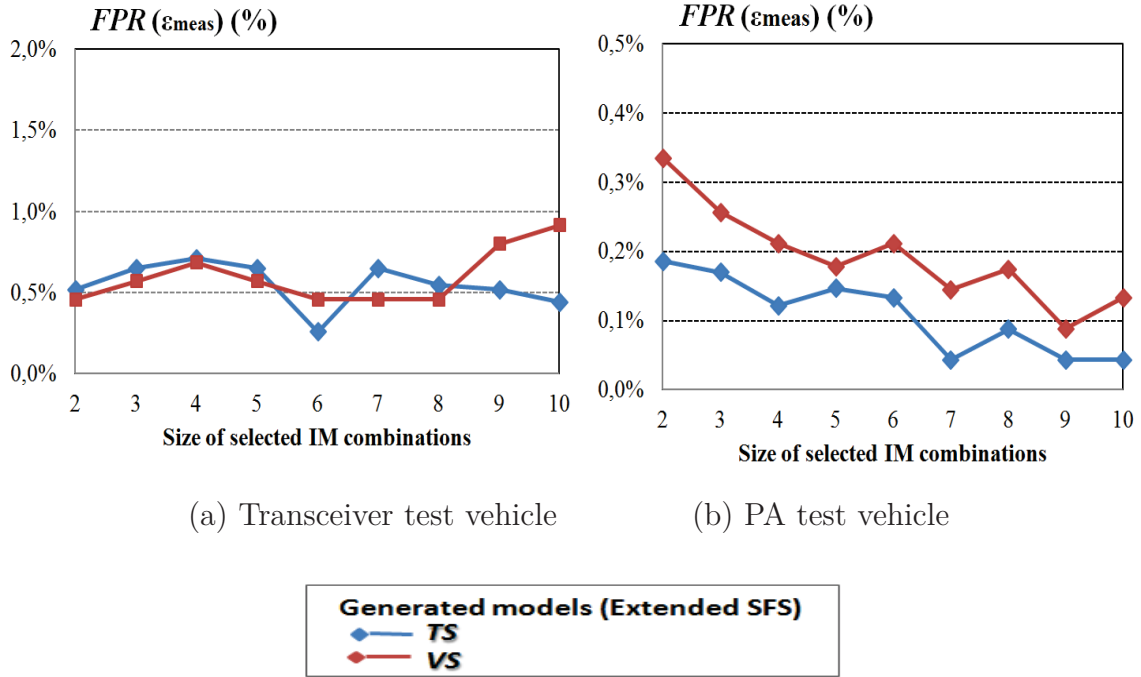


Figure 4.1: Comparison between prediction reliability results evaluated on training and validation sets

4.1 Model redundancy principle

Lack of confidence in test predictions is one of the crucial issues that limit the deployment of the alternate test strategy in the semiconductor industry. An interesting proposal to cope

with this issue is to implement a two-tier test procedure in which confidence estimation is established during the production testing phase: only devices for which confidence is sufficient are predicted using the learned regression models, other devices are directed to another tier where additional testing may be applied to characterize them. In this way, it is expected that most of the devices are evaluated through the low-cost alternate test tier and only a small fraction of devices are evaluated through a more expensive test procedure. As a result, the overall test cost is reduced compared to standard specification testing while accuracy is preserved.

This approach has been explored in [31] [47] in the context of classification-oriented strategy. As illustrated in figure 4.2, it relies on guard-band allocation in the indirect measurement space in order to identify devices for which the alternate test decision is prone to error. More precisely two guard-bands (a "good" one and a "faulty" one) are defined during the training phase that partition the indirect measurement space in three regions: a region that contains only "good" devices, a region that contains only "faulty" devices, and a region interjected in between that contains a mixed distribution. Then during the testing phase, the footprint of the indirect measurement pattern is examined with respect to the learned guard-banded zone: if it falls outside the zone, the device is assigned to the dominant class (either "good" or "faulty"), otherwise if it falls within the zone, the device is deemed suspect to misclassification and it is directed to the second tier when further testing may be applied. Note that this technique has been developed in the context of classification-oriented strategy and necessitates information on passing and failing devices.

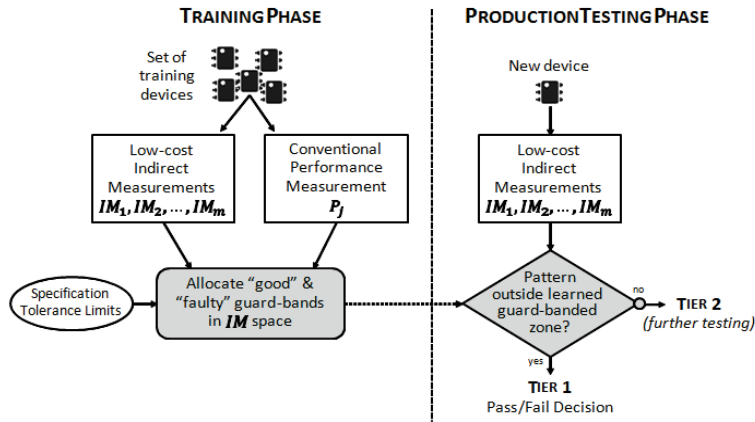


Figure 4.2: Two-tier alternate test synopsis with guard-band allocation

In the context of prediction-oriented strategy, guard-band allocation cannot be applied because test decisions are not taken in the indirect measurement space but in the performance parameter space by comparing predicted values to specification tolerance limits. In this case, another solution has been explored in [58][38][61] based on model redundancy. As illustrated in figure 4.3, the idea is to build several regression models during the training phase and to exploit this redundancy in order establish prediction confidence during the testing phase. More precisely for a given performance to be evaluated, 3 regression models that involve different

combinations of indirect measurements are built during the training phase. Then during the testing phase, confidence estimation is established by checking the consistency between the values predicted by the 3 different models. A device whose performance predictions are similar whatever the regression model used is likely to be properly predicted. In this case, the device can be evaluated by the low-cost alternate test tier. On the contrary, when different models lead to different performance predictions for the same device, we can suspect that at least one of the models does not predict the performance correctly. In this case, prediction is considered as suspect and the device is directed towards the second tier where further testing may be applied.

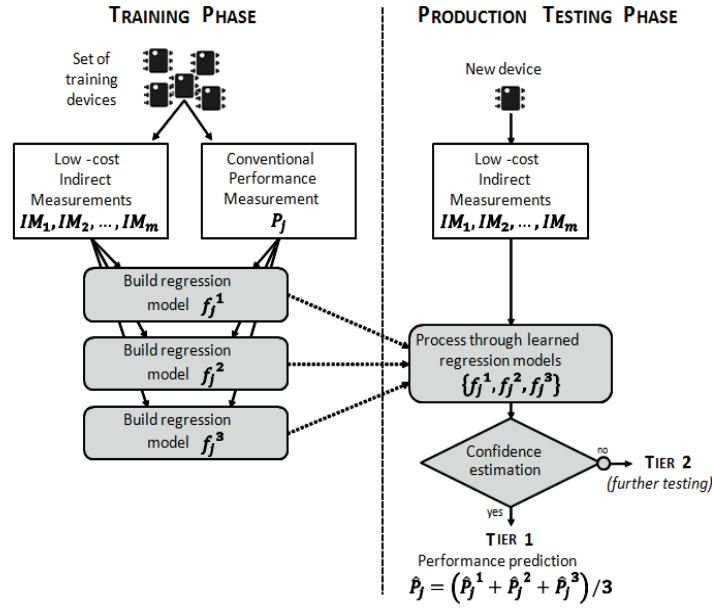


Figure 4.3: Two-tier alternate test synopsis with model redundancy

Figure 4.4 details the procedure for confidence estimation during the testing phase. For every new tested device, the difference between the predicted values is computed for each pair of models and checked against a predefined divergence threshold. If one (or more) of these differences $\Delta \hat{P}_j^{k,l}$ is superior to the divergence threshold ε_{div} , the prediction is considered suspicious and the device is directed to the second tier. On the contrary, if all these differences are inferior to the threshold, it means that there is no discrepancy between the values predicted by the different models and the prediction is considered reliable. In this case, the alternate test tier can be used to evaluate the device performance, which is computed as the mean of the values predicted by the different models:

$$\hat{P}_j = \frac{(\hat{P}_j^1 + \hat{P}_j^2 + \hat{P}_j^3)}{3} \quad (4.1)$$

This strategy based on model redundancy has shown promising results. In particular, it has

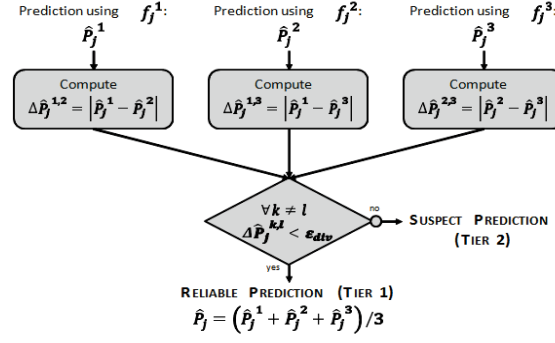


Figure 4.4: Procedure for confidence estimation based on model redundancy

been demonstrated on a real case study that it permits to achieve low rms prediction error, even if training is performed over a small number of instances. However only prediction accuracy was evaluated in the experiments and the choice of the redundant models as well as the choice of the divergence threshold for confidence estimation was done ad-hoc.

4.2 Generic framework

In this chapter, our objective is to develop a generic framework for efficient implementation of the alternate test strategy with model redundancy. This framework should provide a structured methodology for the choice of the redundant models and the choice of an appropriate value for the divergence threshold. It should also permit test efficiency evaluation, not only in terms of prediction accuracy but also in terms of prediction reliability.

4.2.1 Overview

Implementation of model redundancy obviously requires the availability of several prediction models for each device performance. The methodology developed in the previous chapter for multiple model generation is therefore part of the proposed generic framework. As illustrated in figure 4.5, the proposed framework is actually composed of two modules:

- the first one corresponds to the classical implementation of the indirect test strategy and involves (i) outlier removal using the adaptive k-filter, (ii) IM space reduction (optional), and (iii) IM space exploration through extended-SFS search.
- the second one is dedicated to the implementation of model redundancy and involves (i) selection and construction of redundant models, and (ii) exploration of tradeoff between test accuracy and test cost.

In case classical implementation is targeted, only the first module is required. In this case, the IM subset that generates the best model in terms of rms prediction error is retained and

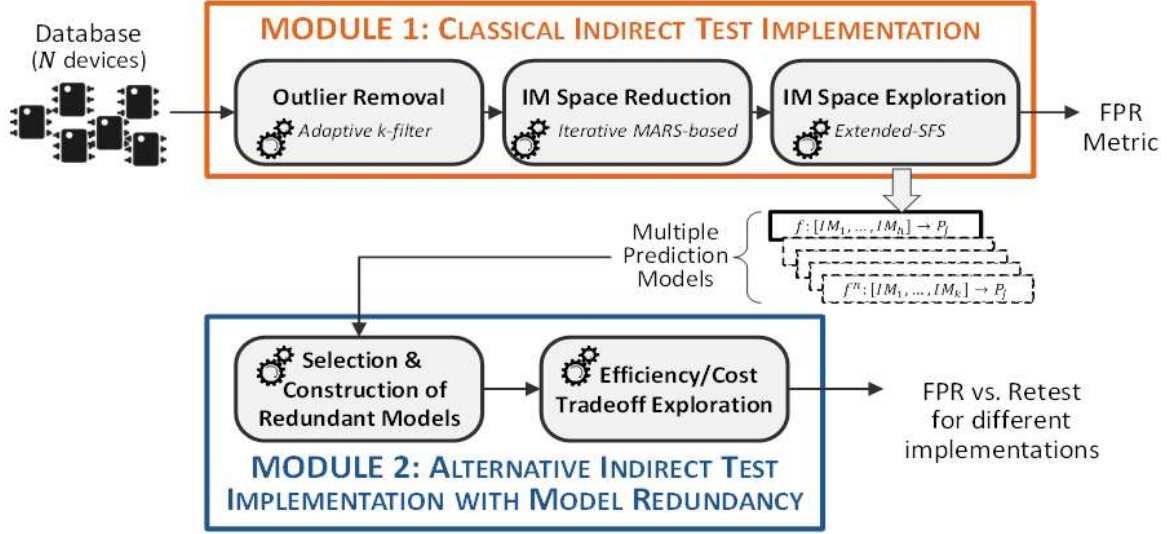


Figure 4.5: Overview of the proposed generic framework

indirect test efficiency is evaluated by computing the FPR metric. In case implementation with model redundancy is targeted, the different prediction models generated by the first module are used as inputs of the second module. Several scenarios are actually implemented for the selection and construction of redundant models and a quantitative evaluation of the performance of different implementations is performed. Details on these steps are given in the following subsections.

4.2.2 Selection and construction of redundant models

From the first module, we have at our disposal several IM subsets of different sizes that can be used to generate prediction models for a given device performance. These subsets have been selected according to the ability of the generated models to accurately represent the relationship between the indirect measurements and the targeted performance for devices of the training set.

Our proposal for the implementation of model redundancy is now to select IM subsets according to the ability of the generated models to correctly predict the performance for devices of the validation set, in particular regarding prediction reliability. More precisely, the learned models are used to perform performance prediction for all devices of the validation set and FPR metric is computed for each model; the 3 models with the best FPR value are then selected, for each size of IM subsets and for the two options of the extended-SFS procedure (parental and non-parental). These models can then be used as redundant models during the production-testing phase in order to establish prediction confidence and estimate the device performance in case of satisfying confidence.

Furthermore in order to reinforce confidence, we investigate an original option, which consists in building the meta-models using ensemble learning [62]. Ensemble learning refers to a collection of methods that learn a target function by training a number of individual learners and

combining their predictions. The resulting model is called ensemble model or meta-model. Numerous empirical and theoretical studies in various domains have demonstrated that ensemble models very often attain higher accuracy than single models. Our objective here is to investigate the use of meta-models in the context of the alternate test strategy.

Various methods exist in the literature for constructing ensembles, e.g. subsampling the training examples, manipulating the input/output features, injecting randomness... In this work we investigate a method based on subsampling the training examples, and more specifically a method based on "cross-validated committees". The principle of this method is illustrated in figure 4.6. The original training set is randomly divided into k partitions and k overlapping training subsets are constructed by dropping out one of these k partitions (typical value for k is 10). Individual models are then learned on each one of these training subsets and the meta-model is obtained by combining predictions of the individual learners with a uniform weighted average.

Practically regarding the implementation of model redundancy with meta-models, our pro-

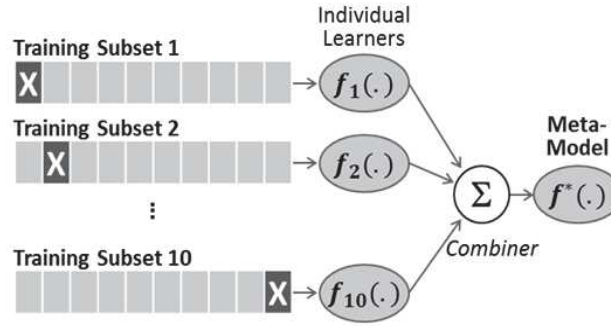


Figure 4.6: Meta-model construction with cross-validated committees ($k=10$)

posal is to consider, for each size of IM subsets, the 3 meta-models corresponding to the 3 standard models selected with respect to their FPR metric. To summarize, different scenarios for selection and construction of redundant models are implemented. First IM subsets are selected based on the individual prediction reliability performance of standard models, considering either the parental or non-parental strategy. Then for each selected subset, two versions of the prediction model are constructed, i.e. a standard model and a meta-model based on cross-validated committees established on training set partitions. These scenarios are summarized in Table 4.1.

Table 4.1: Implemented scenarios for redundant model generation

Model Selection from Extended-SFS	Model Construction
Parental strategy	Standard Models
	Meta-Models
Non-parental strategy	Standard Models
	Meta-Models

4.2.3 Tradeoff exploration: reliability vs. cost

The last phase of the test preparation is related to the choice of the divergence threshold ε_{div} used for confidence estimation and its impact on the tradeoff between test reliability and test cost. Indeed confidence estimation is established by comparing the difference between the predicted values for each pair of redundant values to a pre-defined threshold. If all these differences are inferior to the divergence threshold, the device is predicted using the alternate test tier, otherwise it is directed towards the second tier.

Evidently, the choice of this threshold is crucial since it affects both the number of retested devices and the reliability of test predictions. A strict threshold may exclude many devices from the alternate test tier, resulting in higher test cost. On the contrary a relaxed threshold will direct only few devices to the second test tier, but may allow unreliable predictions for a number of devices. In other words, the choice of this divergence threshold enables the exploration of the tradeoff between test cost and test reliability.

Practically our proposal is to vary the value of the divergence threshold and to evaluate the FPR corresponding to validation devices predicted using the alternate test tier and the percentage of validation devices directed towards the second tier, for each size of selected IM subsets and for each scenario of redundant model generation. As illustrated on the conceptual example of figure 4.7, we can then plot the FPR against the number of retested devices for the different scenarios of redundant model generation. Then we can determine the minimum front, where each point on the optimum curve corresponds to a given scenario and a given divergence threshold. This evaluation is very important because it gives access to quantitative information on the expected performance of alternate test implementation. Based on this information, the test engineer can decide whether alternate test offers sufficient performance for its specific application context and choose the more appropriate implementation.

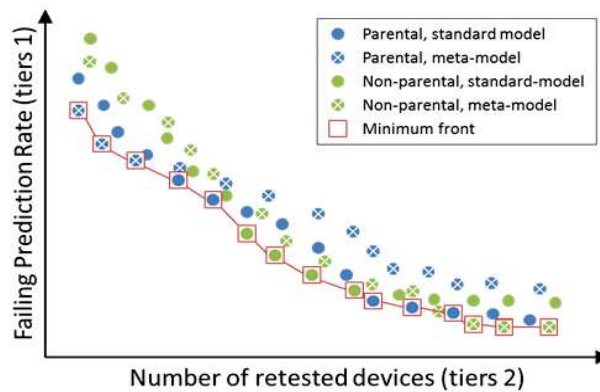


Figure 4.7: Exploration of cost-reliability tradeoff, for a given IM subset size

4.3 Results

The proposed generic framework has been used on our case study, with the objective to implement redundancy using models composed of up to $p = 10$ IMs. The full dataset has been separated in a training set of 2,264 devices and a validation set of 8,943 devices, based on latin-hypercube sampling in order to ensure similar statistical characteristics between both sets.

4.3.1 Selection and construction of redundant models

From the extended-SFS procedure, we have identified for each size of IM combination, 10 subsets using the parental strategy and 10 subsets using the non-parental strategy. The first step for redundant model selection is to use the corresponding standard models to perform performance prediction for all devices of the validation set and evaluate resulting prediction reliability. For each size of IM combination, the 3 models with the best FPR are then retained. As an illustration, figure 4.8 reports prediction reliability results for the 10 standard models generated from the extended-SFS procedure, in case of models built with 3 IMs. From this figure, we observe that the lowest FPR values are obtained for models 1, 9 and 10 with the parental method, and models 5, 9 and 10 with the non-parental method. For the implementation of model redundancy, we will consider those 3 standard models out coming from each strategy. Moreover, we will also consider the corresponding meta-models built using cross-validated committees. For the sake of illustration, FPR values achieved with these meta-models are also reported in figure 4.8. Note that on this example, the use of meta-models does not necessarily yield to better prediction reliability. For instance, the standard model 9 generated with the non-parental method exhibits a better FPR than its corresponding meta-model.

The same process is used to select redundant models with different IM subset sizes. Results are summarized in figure 4.9, which reports intrinsic model accuracy (evaluated on training set) together with prediction accuracy and reliability (evaluated on validation set) for redundant models generated by the different scenarios.

Two main comments arise from these graphs. First, regarding the use of ensemble learning, there is no evidence of the benefit of this option. Indeed whatever the considered metric, generated meta-models can have lower or higher performances than their corresponding standard models depending on the size of selected IM subsets. Yet, performances remain in the same range than the ones achieved with standard models. We will therefore consider this option for the implementation of model redundancy. Then the second comment is more general and concerns the impact of the number of IMs used to build the models. While the rms error evaluated on the training set slightly decreases when increasing the number of IMs, this is not manifest when looking at the rms error evaluated on the validation set. There is therefore no particular interest in selecting a high number of indirect measurements in terms of prediction accuracy. In contrast the decreasing trend is clearly visible when looking at the FPR metric, which suggests that improved reliability might be attained by using redundant models built with a high number of IMs.

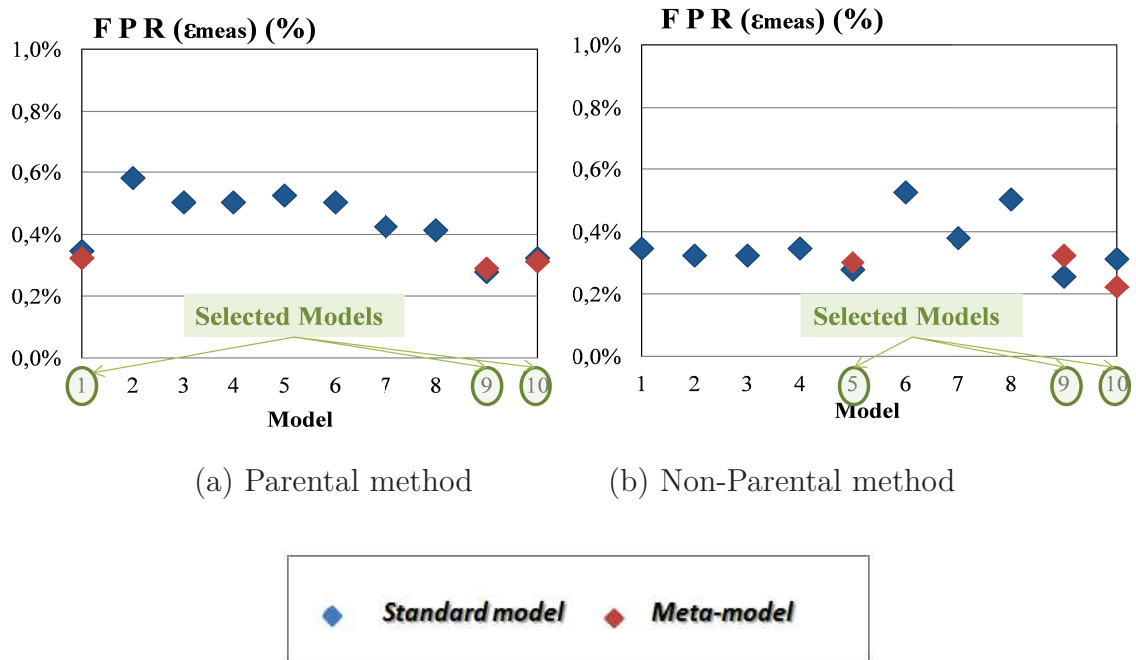


Figure 4.8: Prediction reliability for models built with 3 IMs (models generated from extended-SFS)

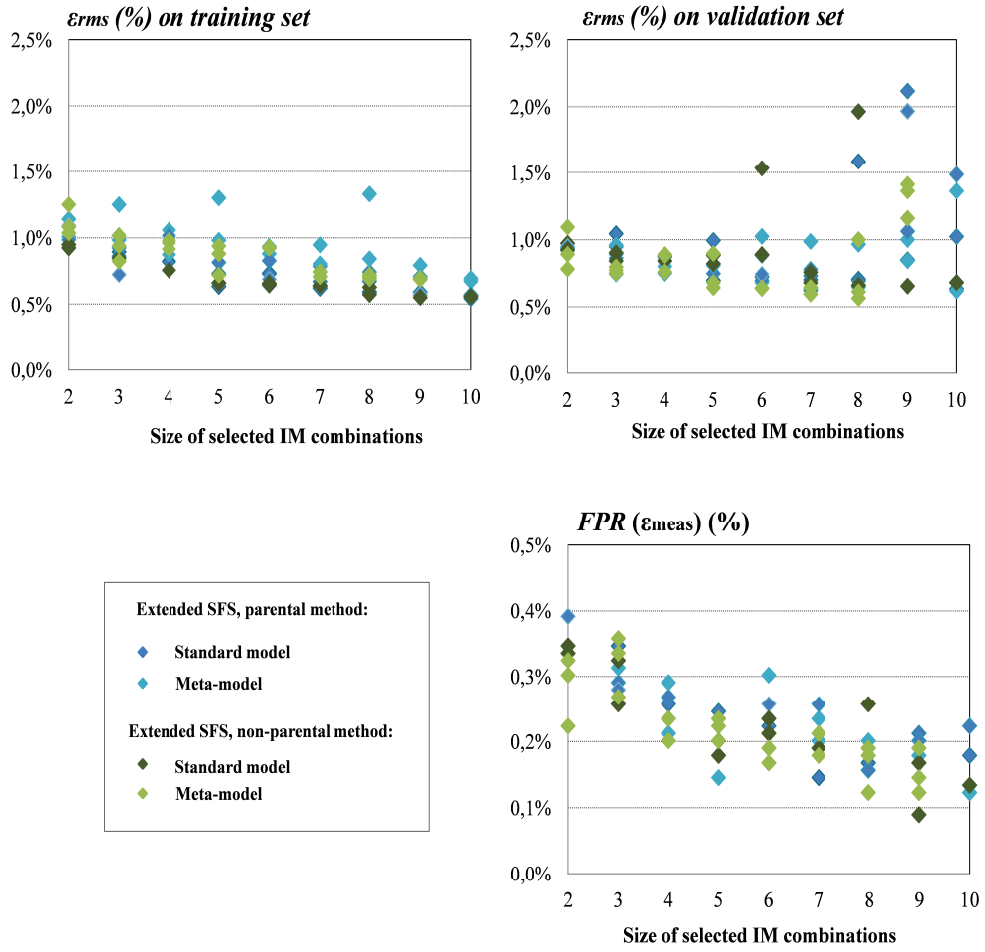


Figure 4.9: Accuracy and reliability metrics for redundant models vs. size of selected IM subsets

4.3.2 Tradeoff between test cost and test reliability

The second phase of the procedure is to explore the tradeoff between test cost and test reliability for the different possible implementations of model redundancy. More precisely, the two-tier test flow is applied for each size of selected IM subsets and each scenario of redundant model generation, considering different values of the divergence threshold. The FPR corresponding to devices directed to the alternate test tier and the percentage of devices directed to the second test tier (Retest) are computed with respect to the value of the divergence threshold. From this evaluation, we can then plot the FPR against Retest, which corresponds to the reliability-cost tradeoff for each possible implementation.

This procedure is illustrated in figure 4.10 for three sizes of selected IM subsets, i.e. 3, 5 and 10. From these graphs, we can note:

- The lower the divergence threshold, the lower the FPR achieved for devices evaluated by the alternate test tier (cf. figure 4.10.a). This clearly illustrates the pertinence of model redundancy which permits to enhance prediction reliability.
- The counterpart of prediction reliability improvement resides in the number of devices directed to the second tier for further testing: the lower the divergence threshold, the higher the percentage of devices directed to the second tier (cf. figure 4.10.b), and therefore the higher the global cost of the two-tier test flow.
- Finally regarding the reliability-cost tradeoff and more specifically the influence of the different scenarios for redundant model generation, there is no evident trend of one scenario being better than the other. This actually depends on both the acceptable level for the percentage of devices directed to the second tier and the size of selected IM subsets (cf. figure 4.10.c). For instance, considering models built with 3 IMs, the best performance in terms of prediction reliability is achieved using meta-models generated from the non-parental strategy in case of 5% acceptable Retest level, and meta-models generated from the parental strategy in case of 10% acceptable Retest level.

To further illustrate this point, figure 4.11 presents the FPR achieved by the different implementations of model redundancy, for every size of selected IM subsets and for 3 different values of the percentage of devices directed to the second tier (Retest equal to 1%, 5% and 10%). These results confirm previous observation, i.e. there is not a single scenario of redundant model generation that leads to superior performance but results depend on both the size of selected IM subsets and acceptable Retest level. In order to optimize the performance of model redundancy implementation, it is therefore important to include the different scenario in the generic framework and to consider the minimum front for each size of selected IM subsets.

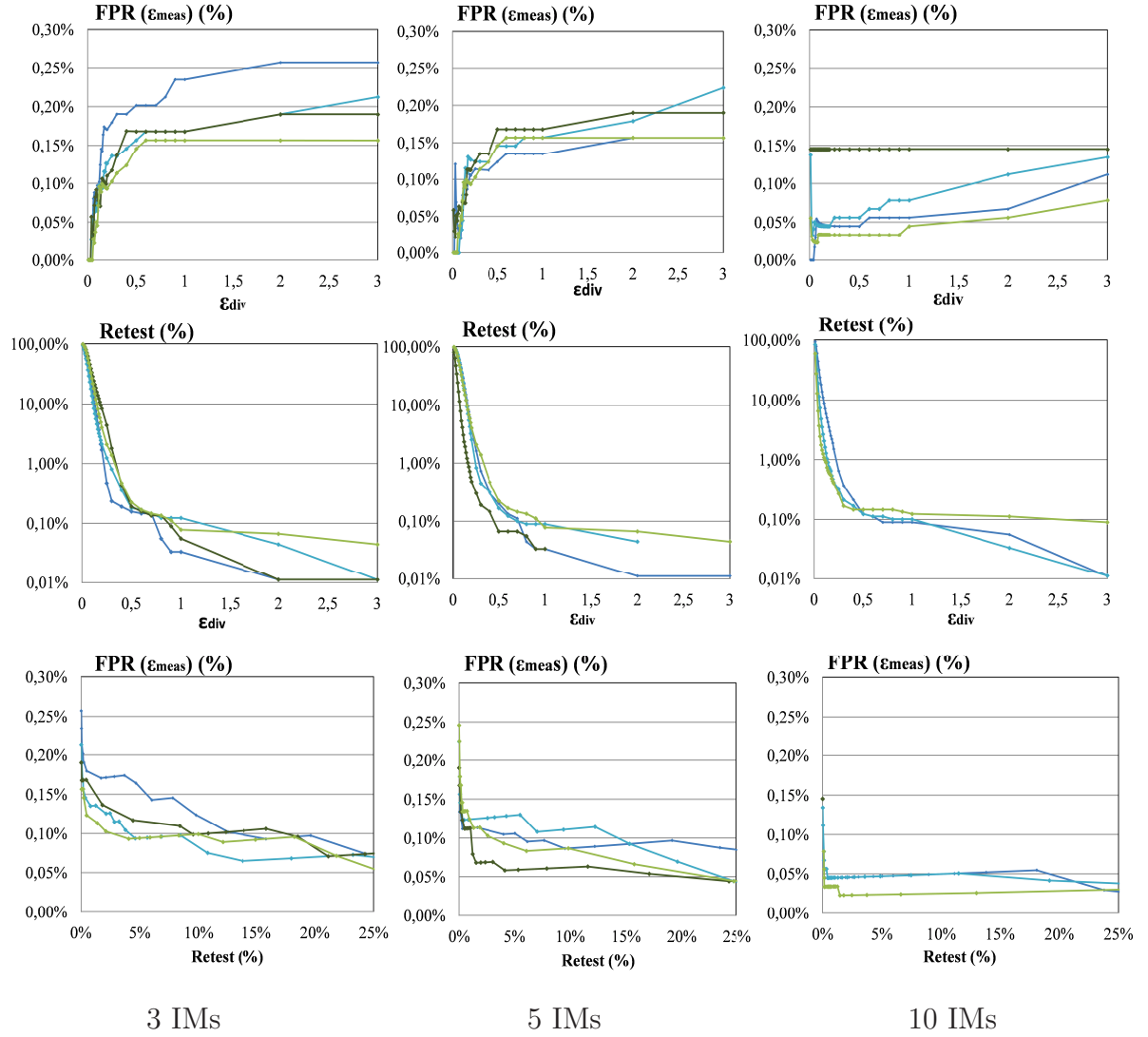


Figure 4.10: Evaluation of different model redundancy implementations: (a) FPR vs. divergence threshold, (b) Retest vs. divergence threshold, (c) FPR vs. Retest

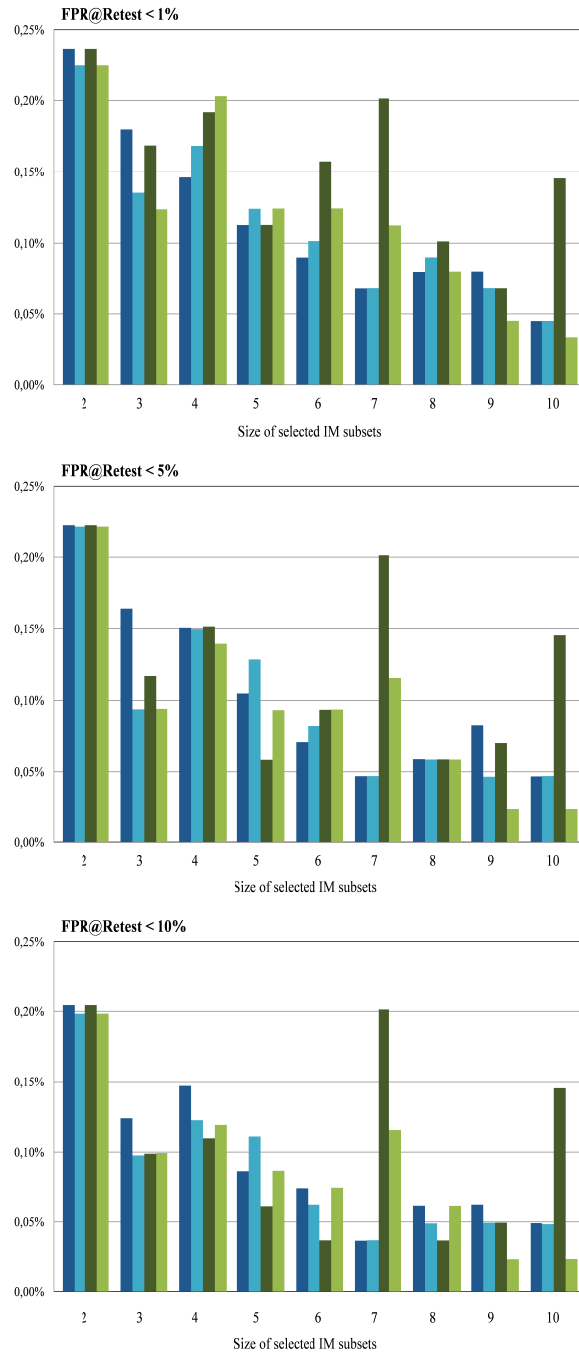


Figure 4.11: Prediction reliability achieved by different implementations of model redundancy, for 3 values of acceptable Restest level

Global results of cost-reliability tradeoff exploration are summarized in figure 4.12 which reports the minimum front of FPR obtained from the different selection scenarios with respect to the percentage of devices directed to the second tier, for the different sizes of selected IM subsets. Each point on the optimum curves corresponds to a given scenario and a given divergence threshold.

This graph contains all the relevant information to help the test engineer choosing the more

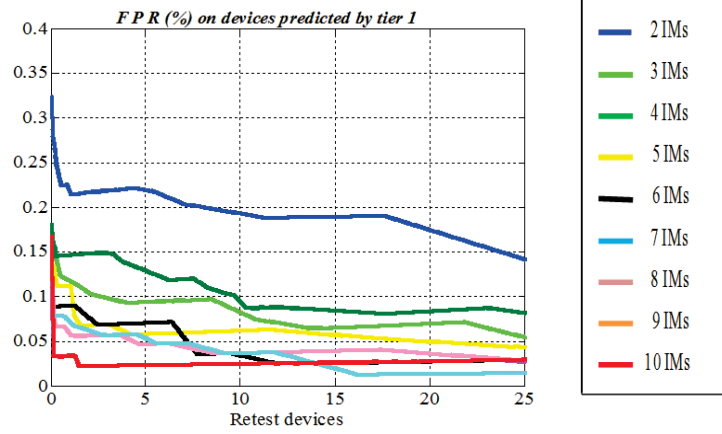


Figure 4.12: Trade-off between test cost and test reliability for different sizes of selected IM subsets (minimum front of FPR obtained from the different scenarios of redundant model generation)

appropriate tradeoff for its specific application context. This graph also clearly illustrates the influence of the number of IMs used to build the redundant models. The general trend is that the higher this number, the lower the FPR that can be reached for a given percentage of devices directed to the second tier. Note that for this case study, interesting performances can be achieved. In particular it is possible to obtain an FPR below 0.05%, which means that more than 99.95% of the devices evaluated by the alternate test tier can be predicted with an error lower than the classical measurement repeatability error. This corresponds to a significant improvement of test reliability compared to the conventional implementation of alternate test with a single prediction model, which permits to reach only an FPR of about 0.2%. Moreover such a low FPR can be obtained with only a small fraction of devices directed to the second tier, incurring very limited test overhead (e.g. less than 0.15% of devices using redundant models built with selected subsets of 10 IMs).

4.4 Summary

In this chapter, we have presented a generic framework for efficient implementation of alternate test with model redundancy. This framework includes the procedures developed in previous chapters, i.e. outlier removal, IM space reduction and multiple IM subset selection through extended-SFS. It is then complemented by two additional procedures dedicated to model redundancy implementation: (i) selection and construction of redundant models and (iii) exploration of cost-reliability tradeoff depending on the choice of the divergence threshold for confidence estimation. Different options have been investigated for IM space exploration and construction of redundant models. The proposed framework permits to select the more appropriate scenario and to have a quantitative evaluation of the alternate test performance in terms of prediction reliability.

The use of the proposed framework has been illustrated on a practical case study for which we have experimental test data. Results clearly demonstrate the benefit of implementing model redundancy, which permits to achieve very low failing prediction rates with only few devices directed to a second tier for further testing. More generally, this framework constitutes an essential element to guide the test engineer regarding practical aspects of alternate test implementation.

Conclusion

INDIRECT test relies on low-cost indirect measurements to estimate RF performances. The fundamental principle is to learn during a training phase regression models that map indirect measurements to the RF performances; these models are then used during the testing phase to predict the RF performances of new manufactured devices using only the indirect measurements. Despite the substantial test cost reduction offered by this strategy and a number of promising results reported in the literature on various case studies, its deployment in industry is today limited. There are several reasons for this and among them, the problem of how to evaluate the efficiency of alternate test and which confidence can be granted to indirect test predictions are major challenges.

The work presented in this PhD manuscript introduces a generic methodology for indirect test implementation, addressing these two challenges. This methodology is illustrated using two test vehicles (a Power Amplifier and an RF Transceiver) for which we have experimental test data provided by NXP Semiconductors.

Considering the first challenge "Evaluation of alternate test efficiency ", we have first performed a comparative analysis of several existing IM selection strategies, which has not been reported before in the literature. We have investigated four different IM strategies, pertaining to both filter (i.e. Pearson and Brownian distance correlations) and wrapper (i.e. MARS and SFS methods) categories. Moreover, we have also investigated the influence of several parameters such as the number of IMs used to perform prediction of a device performance, the size of the training set used during the selection procedure and the size of the training set used for the model construction. Efficiency has been evaluated using classical accuracy metrics such as rms and maximal prediction errors, but also in terms of prediction reliability. For this purpose, we have defined a new metric "Failing Prediction Rate" which permits to quantify the proportion of devices that are predicted with an error below the conventional measurement uncertainty. Results have clearly revealed that finding the set of most appropriate IMs is strongly dependent on the case study. We have shown that with the appropriate selection strategy, good prediction reliability can be achieved using only a limited number of indirect measurements. In particular, for both test vehicles, we found out that 99.5% of the devices can be accurately predicted with an error lower than the measurement repeatability error using only three indirect measurements.

Following this first study, we have focused more specifically on test efficiency improvement

and we have developed a novel solution for the selection of pertinent indirect measurements. This solution relies on an extended-SFS algorithm that allows of more thorough exploration of the IM space, with the objective to (i) possibly identify a better model than with the classical SFS algorithm and (ii) generate several accurate prediction models. In case of IM space of high dimensionality, a preliminary phase "IM-space reduction" is performed to deal with the computational time issue in the context of an extended exploration. We have proposed different IM space reduction solutions (i.e. reduced IM space from: PCA, Pearson correlation and MARS built-in-self-selection) in order to reduce the number of promising candidates from the full IM space. Then regarding IM space exploration, two options (i.e. Extended SFS-Parental and Extended SFS-Non Parental) have been proposed. Obtained results have demonstrated that both options permit to generate several models with performances better or equal than the ones of the single model generated by the classical SFS strategy, in terms of both prediction accuracy and reliability. In case of a conventional implementation of the indirect test, this methodology offers an improved test efficiency by selecting the best model generated by the extended SFS algorithm for each device performance. Moreover the availability of several efficient models for each device performance is actually benefic to address the confidence issue.

In relation with this latter "Enhancement of alternate test confidence", we have presented a generic framework for efficient implementation of indirect test with model redundancy. This framework includes the previous methodology for multiple model generation and complements it with (i) the selection and construction of redundant models through the extended-SFS and (ii) the exploration of tradeoff between test accuracy and test cost. Different options are implemented for the selection and construction of redundant models, and in particular the construction of meta-models using ensemble learning. The exploration of cost-reliability tradeoff is then realized for each scenario depending on the choice of the divergence threshold for confidence estimation. The proposed framework permits to select the more appropriate scenario with respect to a given application context. Results have demonstrated the benefit of implementing model redundancy, which permits to significantly improve the indirect test confidence with only few devices directed to a second tier for further testing. More generally, this framework constitutes an essential element to guide the test engineer regarding practical aspects of alternate test implementation.

This work opens interesting perspectives concerning alternate test 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 and data mining tools in order to fix the appropriate scenario for each case study. Another direction is to study the impact of manufacturing process shift during the production test phase on the predictive models.

Related publications

- [IJ.1] S. Larguech et al., "Efficiency evaluation of analog/RF alternate test: comparative study of indirect measurement selection strategies," (Microelectronics Journal), p.1091-1102, 2015
- [IC.1] S. Larguech et al., "A Framework for Efficient Implementation of Analog/RF Alternate Test with Model Redundancy," (ISVLSI), p.621-626, 2015
- [IW.1] S. Larguech et al., "Evaluation of indirect measurement selection strategies in the context of analog/RF alternate testing," Proc. Latin-American Test Workshop (LATW), p.1-6, 2014
- [IW.2] S. Larguech et al., "A Generic Methodology for Building Efficient Prediction Models in the Context of Alternate Testing," (IMSTW), p.1-6, 2015
- [IW.3] H. AYARI, F. Azais, S. Bernard, V. Kerzerho, M. Comte, S. Larguech, M. Renovell, "Investigations on alternate analog/RF test with model redundancy," Statistical Test Methods Workshop (STEM), 2014

Bibliography

- [1] Sébastien Darfeuille and Christophe Kelma. Production test of an rf receiver chain based on atm combining rf bist and machine learning algorithm. In *Circuit Theory and Design (ECCTD), 2011 20th European Conference on*, pages 653–656. IEEE, 2011.
- [2] Martin Dresler. Technique to detect rf interface and contact issues during production testing. In *2006 IEEE International Test Conference*, 2006.
- [3] Erdem S Erdogan and Sule Ozev. A multi-site test solution for quadrature modulation rf transceivers. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 30(9):1421–1425, 2011.
- [4] Abhijit Chatterjee and Naveena Nagi. Design for testability and built-in self-test of mixed-signal circuits: A tutorial. In *VLSI Design, 1997. Proceedings., Tenth International Conference on*, pages 388–392. IEEE, 1997.
- [5] Andrej Rumiantsev and Ralf Doerner. Rf probe technology: History and selected topics. *Microwave Magazine, IEEE*, 14(7):46–58, 2013.
- [6] Pramodchandran N Variyam and Abhijit Chatterjee. Enhancing test effectiveness for analog circuits using synthesized measurements. In *VLSI Test Symposium, 1998. Proceedings. 16th IEEE*, pages 132–137. IEEE, 1998.
- [7] Pramodchandran N Variyam, Sasikumar Cherubal, and Abhijit Chatterjee. Prediction of analog performance parameters using fast transient testing. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 21(3):349–361, 2002.
- [8] Jay B Brockman and Stephen W Director. Predictive subset testing: Optimizing ic parametric performance testing for quality, cost, and yield. *Semiconductor Manufacturing, IEEE Transactions on*, 2(3):104–113, 1989.
- [9] Haralampos-G Stratigopoulos, Petros Drineas, Mustapha Slamani, and Yiorgos Makris. Rf specification test compaction using learning machines. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 18(6):998–1002, 2010.
- [10] Ram Voorakaranam, Randy Newby, Sasi Cherubal, Bob Cometta, Thomas Kuehl, David Majernik, and Abhijit Chatterjee. Production deployment of a fast transient testing methodology for analog circuits: Case study and results. In *null*, page 1174. IEEE, 2003.

- [11] Selim Sermet Akbay, Achintya Halder, Abhijit Chatterjee, and David Keezer. Low-cost test of embedded rf/analog/mixed-signal circuits in sops. *Advanced Packaging, IEEE Transactions on*, 27(2):352–363, 2004.
- [12] Sasikumar Cherubal, Ram Voorakaranam, Abhijit Chatterjee, John Mclaughlin, Jason L Smith, and David M Majernik. Concurrent rf test using optimized modulated rf stimuli. In *VLSI Design, 2004. Proceedings. 17th International Conference on*, pages 1017–1022. IEEE, 2004.
- [13] Abhishek Halder and Abhijit Chatterjee. Low-cost alternate evm test for wireless receiver systems. In *VLSI Test Symposium, 2005. Proceedings. 23rd IEEE*, pages 255–260. IEEE, 2005.
- [14] Sofiane Ellouz, Patrice Gamand, Christophe Kelma, Bertrand Vandewiele, and Bruno Allard. Combining internal probing with artificial neural networks for optimal rfc testing. In *Test Conference, 2006. ITC’06. IEEE International*, pages 1–9. IEEE, 2006.
- [15] Shalabh Goyal, Abhijit Chatterjee, and Michael Purtell. A low-cost test methodology for dynamic specification testing of high-speed data converters. *Journal of Electronic Testing*, 23(1):95–106, 2007.
- [16] A Chatterjee, Aritra Banerjee, and S Kook. Dynamic specification testing and diagnosis of high precision sigma-delta adcs. *IEEE Design & Test of Computers*, (1):1, 2012.
- [17] Sen-Wen Hsiao, Xian Wang, and Avhishek Chatterjee. Analog sensor based testing of phase-locked loop dynamic performance parameters. In *Test Symposium (ATS), 2013 22nd Asian*, pages 50–55. IEEE, 2013.
- [18] Ian H Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.
- [19] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37, 1996.
- [20] Chinky Gera and Kirti Joshi. A survey on data mining techniques in the medicative field. *International Journal of Computer Applications*, 113(13), 2015.
- [21] Edward Gately. *Neural networks for financial forecasting*. John Wiley & Sons, Inc., 1995.
- [22] Pooja Dixit and Ghanshyam I Prajapati. Machine learning in bioinformatics: A novel approach for dna sequencing. In *Advanced Computing & Communication Technologies (ACCT), 2015 Fifth International Conference on*, pages 41–47. IEEE, 2015.
- [23] HJ Kim, Michael I Jordan, Shankar Sastry, and Andrew Y Ng. Autonomous helicopter flight via reinforcement learning. In *Advances in neural information processing systems*, page None, 2003.
- [24] ZQ John Lu. The elements of statistical learning: data mining, inference, and prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 173(3):693–694, 2010.

- [25] Yuichiro Anzai. *Pattern Recognition & Machine Learning*. Elsevier, 2012.
- [26] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.
- [27] Jerome H Friedman. Multivariate adaptive regression splines. *The annals of statistics*, pages 1–67, 1991.
- [28] Peter L. Flom. Alternative methods of regression when ols is not right. 2015.
- [29] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
- [30] Eunseog Youn and Myong K Jeong. Class dependent feature scaling method using naive bayes classifier for text datamining. *Pattern Recognition Letters*, 30(5):477–485, 2009.
- [31] Haralampos-G Stratigopoulos and Yiorgos Makris. Error moderation in low-cost machine-learning-based analog/rf testing. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 27(2):339–351, 2008.
- [32] Haralampos-G Stratigopoulos, Salvador Mir, Erkan Acar, and Sule Ozev. Defect filter for alternate rf test. In *Test Symposium, 2009 14th IEEE European*, pages 101–106. IEEE, 2009.
- [33] Haralampos-G D Stratigopoulos, Petros Drineas, Mustapha Slamani, and Yiorgos Makris. Non-rf to rf test correlation using learning machines: A case study. In *VLSI Test Symposium, 2007. 25th IEEE*, pages 9–14. IEEE, 2007.
- [34] Haralampos GD Stratigopoulos and Yiorgos Makris. Nonlinear decision boundaries for testing analog circuits. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24(11):1760–1773, 2005.
- [35] Viera Stopjaková, Pavol Malošek, Marek Matej, Vladislav Nagy, and Martin Margala. Defect detection in analog and mixed circuits by neural networks using wavelet analysis. *Reliability, IEEE Transactions on*, 54(3):441–448, 2005.
- [36] Ram Voorakaranam, Selim Sermet Akbay, Soumendu Bhattacharya, Sasikumar Cherubal, and Abhijit Chatterjee. Signature testing of analog and rf circuits: Algorithms and methodology. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 54(5):1018–1031, 2007.
- [37] Ronald L Iman, James M Davenport, and Diane K Zeigler. Latin hypercube sampling (program user’s guide).[lhc, in fortran]. Technical report, Sandia Labs., Albuquerque, NM (USA), 1980.
- [38] Haithem Ayari, Florence Azais, Serge Bernard, Mariane Comte, Vincent Kerzerho, Olivier Potin, and Michel Renovell. On the use of redundancy to reduce prediction error in alternate analog/rf test. In *Mixed-Signals, Sensors and Systems Test Workshop (IMS3TW), 2012 18th International*, pages 34–39. IEEE, 2012.

- [39] Haithem Ayari, Florence Azaïs, Serge Bernard, Mariane Comte, Vincent Kerzerho, and Michel Renovell. Enhancing confidence in indirect analog/rf testing against the lack of correlation between regular parameters and indirect measurements. *Microelectronics Journal*, 45(3):336–344, 2014.
- [40] Syhem Larguech, Florence Azais, Serge Bernard, Vincent Kerzerho, Mariane Comte, and Michel Renovell. Evaluation of indirect measurement selection strategies in the context of analog/rf alternate testing. In *Test Workshop-LATW, 2014 15th Latin American*, pages 1–6. IEEE, 2014.
- [41] JD Wichard, MJ Ogorzalek, and C Merkwirth. Entool-a toolbox for ensemble modelling. In *Europhysics conference abstracts ECA*, volume 27, pages A105–A105. EUROPEAN PHYSICAL SOCIETY, 2003.
- [42] Haithem Ayari. *Indirect Analog/RF IC Testing: Confidence & Robustness improvements*. PhD thesis, Université Montpellier II-Sciences et Techniques du Languedoc, 2013.
- [43] Douglas M Hawkins. *Identification of outliers*, volume 11. Springer, 1980.
- [44] Gordon P Hughes. On the mean accuracy of statistical pattern recognizers. *Information Theory, IEEE Transactions on*, 14(1):55–63, 1968.
- [45] Vladimir Cherkassky and Filip M Mulier. *Learning from data: concepts, theory, and methods*. John Wiley & Sons, 2007.
- [46] Victoria J Hodge and Jim Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126, 2004.
- [47] Nathan Kupp, Petros Drineas, Mustapha Slamani, and Yiorgos Makris. Confidence estimation in non-rf to rf correlation-based specification test compaction. In *Test Symposium, 2008 13th European*, pages 35–40. IEEE, 2008.
- [48] S Larguech, F Azais, S Bernard, M Comte, V Kerzerho, and M Renovell. A generic methodology for building efficient prediction models in the context of alternate testing. In *Mixed-Signal Testing Workshop (IMSTW), 2015 20th International*, pages 1–6. IEEE, 2015.
- [49] S Sermet Akbay and Abhijit Chatterjee. Built-in test of rf components using mapped feature extraction sensors. In *VLSI Test Symposium, 2005. Proceedings. 23rd IEEE*, pages 243–248. IEEE, 2005.
- [50] Louay Abdallah, Haralampos-G Stratigopoulos, Christophe Kelma, and Salvador Mir. Sensors for built-in alternate rf test. In *Test Symposium (ETS), 2010 15th IEEE European*, pages 49–54. IEEE, 2010.
- [51] Manuel J Barragan, Rafaella Fiorelli, Gildas Leger, Adoracion Rueda, and Jose L Huer-tas. Improving the accuracy of rf alternate test using multi-vdd conditions: application to envelope-based test of lnas. In *Test Symposium (ATS), 2011 20th Asian*, pages 359–364. IEEE, 2011.

-
- [52] Haralampos-G Stratigopoulos. Fundamentals of machine learning and its applications in test. IEEE, 2015.
 - [53] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003.
 - [54] Manuel J Barragan and Gildas Leger. Efficient selection of signatures for analog/rf alternate test. In *Test Symposium (ETS), 2013 18th IEEE European*, pages 1–6. IEEE, 2013.
 - [55] Gábor J Székely, Maria L Rizzo, et al. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009.
 - [56] Haithem Ayari, Florence Azais, Serge Bernard, Mariane Comte, Michel Renovell, Vincent Kerzerho, Olivier Potin, and Christophe Kelma. Smart selection of indirect parameters for dc-based alternate rf ic testing. In *VLSI Test Symposium (VTS), 2012 IEEE 30th*, pages 19–24. IEEE, 2012.
 - [57] John Liaperdos, Angela Arapoyanni, and Yiorgos Tsiatouhas. Adjustable rf mixers’ alternate test efficiency optimization by the reduction of test observables. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 32(9):1383–1394, 2013.
 - [58] Haithem Ayari, Florence Azais, Serge Bernard, Mariane Comte, Vincent Kerzerho, Olivier Potin, and Michel Renovell. Making predictive analog/rf alternate test strategy independent of training set size. In *Test Conference (ITC), 2012 IEEE International*, pages 1–9. IEEE, 2012.
 - [59] Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.
 - [60] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer, 2009.
 - [61] Haithem Ayari, Florence Azais, Serge Bernard, Mariane Comte, Vincent Kerzerho, and Michel Renovell. New implementations of predictive alternate analog/rf test with augmented model redundancy. In *Proceedings of the conference on Design, Automation & Test in Europe*, page 131. European Design and Automation Association, 2014.
 - [62] Zhi-Hua Zhou. *Ensemble methods: foundations and algorithms*. CRC Press, 2012.